CLIFF TOKENS: ANALYZING FAILURE TRIGGER TOKENS IN LLM MATHEMATICAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models reach high accuracy in mathematical reasoning, but individual traces on the same problem diverge; some arrive at the correct answer while others fail. Prior work localizes such failures at the step, chunk, or sentence level, or identifies tokens where failure has already occurred. These approaches leave open which token triggers failure. We introduce the **cliff token**, a token at which the estimated probability of reaching the correct answer (success probability) drops beyond an adaptive threshold. Across seven models and three mathematical reasoning benchmarks (GSM1K, MATH500, AIME 2025), cliff tokens act as failure triggers. For incorrect traces containing cliff tokens, we compare resampling immediately before and after the first cliff token. Resampling before it shows higher pass@k at the same sample count. We further introduce a cliff taxonomy of deterministic, uncertain, and sampled-off cliffs, defined by greedy choice and token entropy. Additionally, we show that the three types differ as training signals. Using single-token preference optimization at cliff positions (Cliff-DPO), we find that uncertain and sampled-off cliffs show larger accuracy gains than deterministic cliffs on three evaluation benchmarks. We release token-level rollout data and source code to enable further analysis without regenerating costly rollouts `https://anonymous.4open.science/r/cliff_token_data/`.

1 INTRODUCTION

Large language models (LLMs) have shown strong performance on mathematical reasoning tasks (Wei et al., 2022; Shao et al., 2024; Guo et al., 2025; Hui et al., 2024; OpenAI et al., 2024). Still, individual reasoning traces can fail to reach the correct answer despite the underlying LLM being capable of solving the problem (as reflected in the gap between pass@1 and pass@k) (Wang et al., 2023; Liu et al., 2025b). Such failures do not necessarily stem from a global lack of capability, but from specific, critical tokens that shift a reasoning path toward an incorrect outcome (Lin et al., 2025; Wang et al., 2025; Zhu et al., 2026). Despite their impact, how to identify these tokens and analyze their characteristics remains an open question.

Existing works explore this question from three directions: (i) *Broader granularity:* Several reasoning analysis studies focus on coarse-grained units such as reasoning steps (Ton et al., 2025; Zhang et al., 2026), chunks (Bachmann et al., 2026; Hammoud et al., 2025) or sentences (Bogdan et al., 2025; Lanham et al., 2023), rather than at the token level at which generation occurs. (ii) *Post-failure states:* While Lin et al. (2025) flags tokens where the estimated probability of reaching the correct answer from the current prefix reaches zero, the preceding token that triggers this drop has not been identified. (iii) *Absolute thresholds:* Prior studies often use fixed thresholds for probability drops (e.g., 0.3 in Bachmann et al. (2026) or 0.2 in Abdin et al. (2024)), without distinguishing statistically significant shifts from rollout noise.

To address these gaps, we identify which tokens shift a reasoning trace toward failure and analyze their characteristics. We define the **cliff token**: the token in a reasoning trace where the probability of reaching the correct answer via Monte Carlo rollouts (success probability) drops significantly. For example, in Figure 1 (Left), the highlighted token '7' shifts the trace toward an incorrect factorization of 1,092, inserting an extra factor of 7 where the correct factorization requires 3. The trace is still recoverable before this token, but once '7' is sampled, most continuations lead to incorrect answers

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

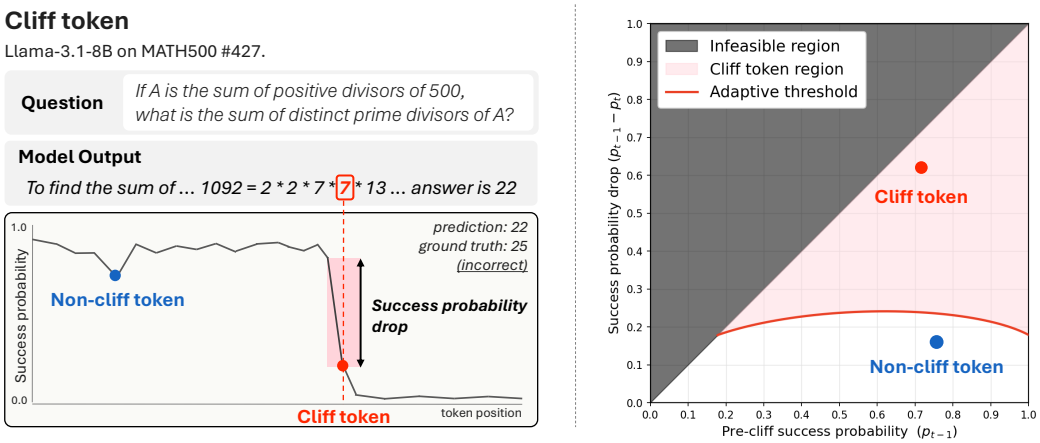


Figure 1: Cliff token identification. (Left) Example reasoning trace where the token '7' produces a drop in success probability, after which the trace proceeds to an incorrect answer. (Right) Cliff-token decision based on the adaptive threshold: the red region denotes drops satisfying the cliff-token criterion. The red point is identified as a cliff token; the blue point is not.

and the success probability collapses. To distinguish statistically significant shifts from sampling noise, we apply an *adaptive threshold* based on a one-sided two-proportion z-test (Figure 1, Right).

Using this framework, we analyze cliff tokens across seven models (`Qwen3-8B`, `Qwen3-4B`, `Qwen3-0.6B` (Yang et al., 2025), `Llama-3.1-8B`, `Llama-3.2-3B`, `Llama-3.2-1B` (Grattafiori et al., 2024), and `Gemma-3-4B` (Gemma Team et al., 2025)) and three benchmarks (GSM1K (Zhang et al., 2024), MATH500 (Lightman et al., 2024), and AIME 2025 (OpenCompass, 2025)). We show that cliff tokens act as failure triggers: resampling immediately before the first cliff token generally yields higher pass@k than after it. We find that cliff tokens do not necessarily coincide with error tokens, and after a cliff token, traces can be recovered as the model self-corrects (Section 4). We identify three types of cliff tokens—*deterministic*, *uncertain*, and *sampled-off cliffs*—based on token entropy and whether the token is the greedy choice (Section 5). We show that the cliff types differ as training signals using single-token preference optimization at cliff positions (Cliff-DPO). Uncertain and sampled-off cliffs show larger accuracy gains than deterministic ones, as the models already assign near-certain probabilities to deterministic cliffs (Section 6). Finally, we release token-level rollout data to support further research on LLM reasoning. Our main contributions can be summarized as follows:

- We formalize the *cliff token* with a z-test based adaptive threshold that separates statistically significant drops in success probability from sampling noise, showing that these tokens act as failure triggers.
- We introduce a *cliff taxonomy—deterministic*, *uncertain*, and *sampled-off cliffs*—showing that the proportion of each type varies across model families and sizes.
- We show that uncertain and sampled-off cliffs are more effective as training signals than deterministic ones under Cliff-DPO, as the models assign near-certain probabilities to deterministic cliffs.

2 Related Work

Localizing reasoning failures. Prior work has studied reasoning errors in LLM traces at coarser levels of granularity, such as steps, chunks, and sentences (Lightman et al., 2024; Wang et al., 2024; Luo et al., 2024; Hwang et al., 2024; Bachmann et al., 2026; Bogdan et al., 2025; Xiong et al., 2025). However, a single token can shift a reasoning trace toward failure. Such tokens can be identified by a drop in the estimated probability of reaching the correct answer after the token is generated. However, existing definitions may miss tokens that shift a trace toward failure. For example, Lin et al. (2025) identify tokens only when the success probability reaches zero, but we show that success

probability can drop sharply while remaining above zero. Abdin et al. (2024) use a fixed threshold, which does not incorporate sampling noise. We instead identify tokens where the success probability drops sharply, even if it remains above zero, using a statistically motivated adaptive threshold.

Token-level preference optimization. Beyond failure identification, token-level signals have been used to extend DPO (Rafailov et al., 2023). Zeng et al. (2024), Liu et al. (2025a), and Zhu et al. (2025) introduce token-level DPO objectives that incorporate KL regularization, importance weights, or per-token reward guidance. Lin et al. (2025) estimate critical tokens using contrastive estimation and use them as token-level weights in the DPO loss. Zhang et al. (2025) concentrate preference optimization on tokens within error-prone regions in code generation. Abdin et al. (2024) construct single-token DPO pairs at pivotal positions identified by recursive subdivision. We test *Cliff-DPO*, targeting cliff positions for preference optimization, and find that it improves reasoning performance with effectiveness varying across cliff types (Section 6).

3 CLIFF TOKEN

3.1 CLIFF TOKEN: DEFINITIONS AND FORMALIZATION

A **cliff token** is a token where the success probability drops significantly when it is generated (see Figure 1, Left). Success probability[1] is the probability of reaching the correct answer conditioned on the reasoning trace generated up to a specific token position t. Formally, given a prompt $\boldsymbol{x}$, a partial reasoning sequence $\boldsymbol{c}_{\leq t}$, and the ground-truth answer y^*, the success probability is defined as follows:

$$P(\boldsymbol{c}_{\leq t}; \boldsymbol{x}) := \mathbb{P}_{(\boldsymbol{c}_{>t}, y) \sim \mathrm{LM}_\theta(\cdot|\boldsymbol{c}_{\leq t}, \boldsymbol{x})}(y = y^*). \tag{1}$$

We empirically estimate success probability, denoted as $\widehat{P}_N$, by executing N rollouts from every token position t and computing the success rate of reaching the ground-truth answer:

$$\widehat{P}_N(\boldsymbol{c}_{\leq t}; \boldsymbol{x}) := \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{y^{(n)} = y^*\}} \quad \text{where } \left(\boldsymbol{c}_{>t}^{(n)}, y^{(n)}\right) \sim \mathrm{LM}_\theta(\cdot|\boldsymbol{c}_{\leq t}, \boldsymbol{x}). \tag{2}$$

With a finite number of rollouts, the estimated success probabilities are subject to sampling noise (Sun et al., 2026; Bigelow et al., 2026; Zhang et al., 2026). Thus, identifying success probability fluctuations based on absolute changes is statistically unreliable. While increasing the number of rollouts could mitigate this variance, it is computationally prohibitive given the quadratic complexity of success probability estimation (see Appendix C.1). To reliably identify cliff tokens under these constraints, we use an *adaptive threshold* based on one-sided two-proportion z-test at a 95% confidence level. Let $\widehat{p}_t := \widehat{P}_N(\boldsymbol{c}_{\leq t}; \boldsymbol{x})$ denote the estimate of success probability at position t. A token at position t is identified as a cliff token if the success probability drop $\Delta_t = \widehat{p}_{t-1} - \widehat{p}_t$ satisfies the following:

$$\Delta_t > \tau + 1.645 \cdot \mathrm{SE}_t, \quad \text{where} \quad \mathrm{SE}_t = \sqrt{\frac{\widehat{p}_{t-1}(1-\widehat{p}_{t-1})}{N} + \frac{\widehat{p}_t(1-\widehat{p}_t)}{N}}. \tag{3}$$

This adaptive threshold combines the base threshold τ and a statistical margin $1.645 \cdot \mathrm{SE}_t$. The margin dynamically accounts for the variance in the probability estimation and reduces false-positive detections in high-variance regions.

3.2 EXPERIMENT SETTINGS

Models. We use seven instruction-tuned models across different scales and model families: `Qwen3-8B/4B/0.6B`, `Llama-3.1-8B-Instruct`, `Llama-3.2-3B/1B-Instruct`, and `Gemma-3-4B-it`. For generating stem traces, we use the temperature[2] T, top-p, and top-k settings recommended by each model provider. See Appendix D for inference hyperparameters.

[1] This rollout-based quantity has been referred to as *potential* (Bachmann et al., 2026), *correctness score* (Lin et al., 2025), and *success probability* (Abdin et al., 2024). We adopt the last term.

[2] `Qwen3` ($T = 0.7$, non-thinking mode), `Llama-3` ($T = 0.6$), and `Gemma-3` ($T = 1.0$).

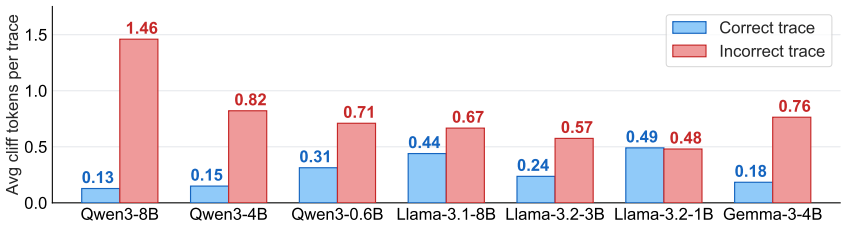


Figure 2: Average cliff tokens per trace, in correct vs. incorrect cases. Aggregated across GSM1K, MATH500, and AIME 2025.

Datasets. We use three mathematical reasoning datasets: GSM1K, MATH500, and AIME 2025. Following Lin et al. (2025); Bogdan et al. (2025); Bigelow et al. (2025), we use subsampling to reduce the computational cost of token-wise rollouts. We randomly sample 100 problems each from GSM1K and MATH500 and use the full set of 30 problems from AIME 2025. Even with this subsampling, success probability estimation required 4,047 A100 (80GB) GPU-hours for processing 1,610 model–problem pairs (7 models, 230 problems). See Appendix C.2 for the empirical GPU-hour breakdown. We publicly release the token-level rollout data used to estimate success probabilities so that other researchers can analyze cliff tokens without regenerating these computationally expensive rollouts.

Threshold. We set the base threshold $\tau = 0.1$, following Bachmann et al. (2026), who define monotonicity in reasoning as a maximum success probability drop of 0.1 between consecutive steps. We also considered setting $\tau = 0.2$ or higher (which shifts the adaptive range to $[0.294, 0.337]$; see Appendix A), but preliminary analyses showed this was too strict and missed many clear failure-triggering tokens.

3.3 Cliff token identification

Generating stem traces. For each problem $\boldsymbol{x}$ and model LM_θ, we use a common zero-shot prompt (Appendix D.4) to sample exactly one complete response, which we call the stem trace. We neither select the best of multiple responses nor edit the stem trace, and we keep it regardless of whether its final answer is correct.

Rollout protocol. To estimate the success probability, we perform $N = 64$ rollouts at each evaluated position. To further reduce computational cost, we apply an early-termination heuristic inspired by similar strategies in reinforcement learning with verifiable rewards (Xu et al., 2026; Xing et al., 2026): if the success probability remains 0 for 20 consecutive tokens, we consider the reasoning trace irrecoverable and truncate all subsequent rollouts for that sequence. We also set dataset-specific maximum token lengths depending on their difficulty (see Appendix D). Using the adaptive threshold defined in Section 3.1, we identify cliff tokens from drops in success probability between adjacent token prefixes.

4 Do cliff tokens trigger reasoning failures?

4.1 Cliff tokens occur more often in incorrect traces

Figure 2 shows that cliff tokens occur more frequently in incorrect traces than in correct traces for most models, with `Llama-3.2-1B` as the only exception. Among incorrect traces, `Qwen3-8B` generates the most cliff tokens per trace on average, nearly twice as many as the next-highest model. The lack of separation between correct and incorrect traces for `Llama-3.2-1B` reflects two causes. First, 52.5% of its incorrect traces never reach the minimum adaptive threshold (0.180), so cliff tokens cannot exist in incorrect traces by definition. Second, `Llama-3.2-1B` fails to retain successful prefixes due to low reasoning ability, as observed in highest cliff token rates in correct traces (0.49/trace).

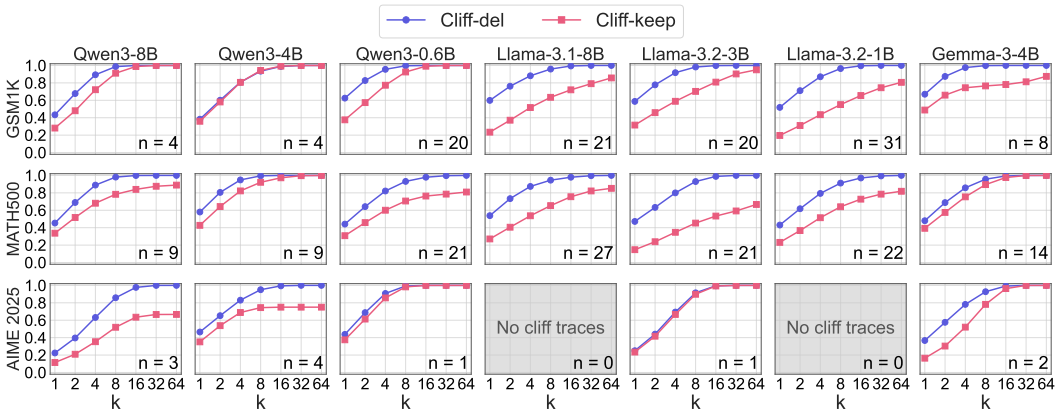


Figure 3: Cliff-del and Cliff-keep pass@k results on incorrect traces. The gap between Cliff-del and Cliff-keep across pass@k shows that removing a cliff token can restore reasoning performance. n denotes the number of incorrect traces containing at least one cliff token. Gray panels mean no cliff tokens in that setting. See Appendix G for detailed results on both incorrect and correct traces.

4.2 Removing a single cliff token recovers reasoning traces

We investigate the effect of cliff tokens on reasoning traces by observing how model decoding behaves when these tokens are kept versus removed. Specifically, we focus on incorrect traces that contain at least one cliff token. To ensure a fair comparison and prevent traces with multiple cliff tokens from being overrepresented in the results, we restrict our evaluation to the *first* occurring cliff token per trace. Let c_{t^*} denote this first cliff token. Using newly sampled continuations independent of the detection rollouts, we evaluate pass@k under two prefix conditions:

- **Cliff-del:** Resampling from the prefix $\boldsymbol{x} \oplus \boldsymbol{c}_{<t^*}$, excluding the cliff token c_{t^*}. This setup allows us to test whether the model can diverge from the failure trace when the cliff token is deleted.
- **Cliff-keep:** Resampling from the prefix $\boldsymbol{x} \oplus \boldsymbol{c}_{\leq t^*}$, including the cliff token c_{t^*}. This setup forces the model to continue decoding directly from the point of the success probability drop.

Figure 3 shows the pass@k results for both the Cliff-del and Cliff-keep setups. Cliff-del shows higher point estimates than Cliff-keep at every k until both reach 1.0, except for `Qwen3-4B` on GSM1K at $k = 8, 16, 32$, where Cliff-keep is only marginally higher. This gap is particularly notable at small k (e.g., for `Llama-3.1-8B`, pass@1 is 0.60 for Cliff-del vs. 0.23 for Cliff-keep on GSM1Kc), showing that removing the cliff token leads to a substantial recovery in reasoning performance without requiring extensive resampling. Furthermore, the Cliff-del pass@64 rate reaches 1.0 across all evaluated panels, indicating that these traces are solvable when the cliff token is removed. Conversely, the Cliff-keep pass@64 rates remain between 0.67 and 1.00 across panels with cliff tokens, indicating that in some panels, even 64 resamples are insufficient to recover the trace once the cliff token is fixed. These results suggest that cliff tokens act as triggers of reasoning failures.

4.3 Cliff tokens are local failure triggers in recoverable dynamics

In contrast to critical tokens, which identify positions where success probability has already collapsed to zero (Lin et al., 2025), cliff tokens capture local shifts toward failure that are not necessarily irreversible. In Figure 4, a correct `Qwen3-4B` trace for AIME 2025 Problem 1 contains two cliff tokens. Success probability drops at the first cliff token (0.69 to 0.34) and the second (0.44 to 0.16), then recovers to 0.80 once the model writes "But wait" and starts to correct its arithmetic. More broadly, among the 358 traces containing at least one cliff token, 47.2% contain multiple cliff tokens and 32.4% still reach the correct answer (see Appendix F). These results reflect non-monotonic reasoning dynamics: success probability can increase after a cliff token as the model verifies, backtracks, or self-corrects. Furthermore, in Figure 4, neither cliff token coincides with the actual arithmetic error itself (the miscomputed value 443). This aligns with findings that critical tokens do not necessarily coincide with human-annotated error tokens (Lin et al., 2025).

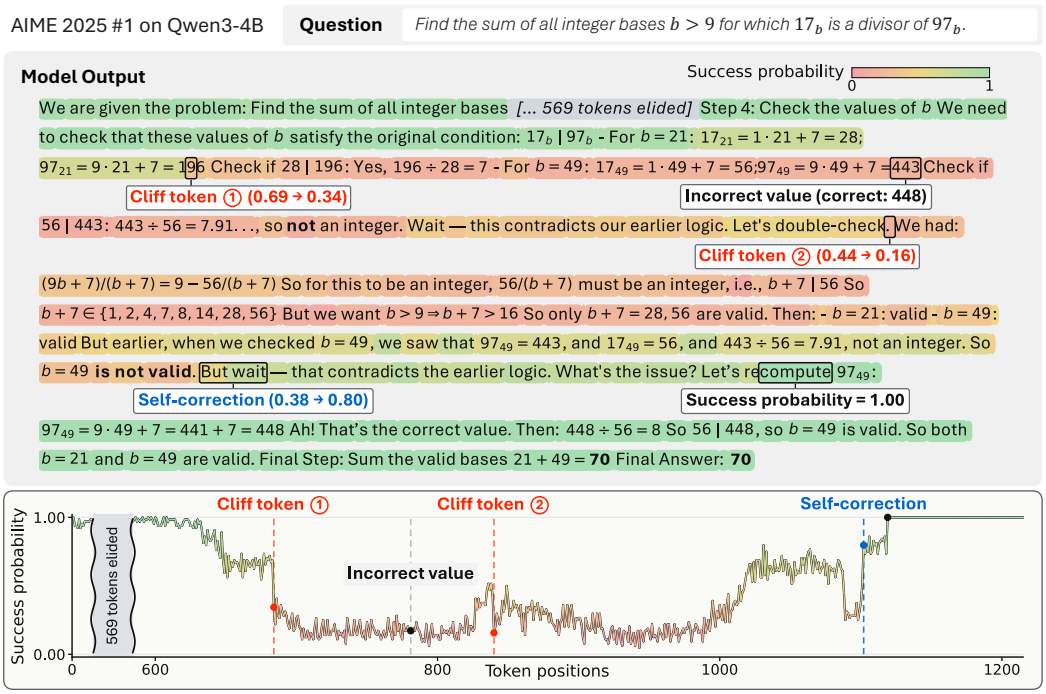


Figure 4: An example reasoning trace that reaches the correct answer despite containing two cliff tokens (`Qwen3-4B`, AIME 2025 Problem 1). Both cliff tokens produce a drop in success probability. The semantic error point (calculation error; 443) is different from the cliff tokens. Success probability recovers after the model writes "But wait" and corrects its reasoning.

5 Cliff Taxonomy

5.1 Definition of Cliff Taxonomy

Are all cliff tokens equal? To analyze how the token sampling process affects success probability, we characterize cliff tokens along two criteria: whether the sampled token is the greedy token, and the entropy H of the next-token distribution at that position. First, one might suspect that cliff tokens happen because the sampled token was not the top choice; we find that many of the cliff tokens are still greedy (38%–82%); see Appendix I. Second, while the entropy of ordinary reasoning tokens is concentrated near $H \approx 0$, cliff tokens show lower density in this low-entropy regime and a heavier-tailed distribution; see Appendix H. This indicates that many cliff tokens occur under local uncertainty. These observations motivate the following cliff taxonomy.

To formalize this taxonomy, we first define these two criteria. Let $p_t(v) = p(v \mid x, c_{<t})$ denote the next-token probability of $v \in \mathcal{V}$ at token position t. The sampled token c_t is greedy if $c_t \in \arg\max_{v \in \mathcal{V}} p_t(v)$ and non-greedy otherwise. The token entropy at position t is

$$H_t = -\sum_{v \in \mathcal{V}} p_t(v) \log p_t(v). \tag{4}$$

We then set an entropy threshold below which we consider a position near-deterministic. Rather than choosing an entropy value directly, we anchor the threshold to the probability of the greedy token $p_1 = 0.99$. Among all distributions whose greedy probability is p_1, the minimum token entropy is the binary entropy

$$H_b(p_1) = -p_1 \log(p_1) - (1 - p_1) \log(1 - p_1), \tag{5}$$

which depends only on p_1, making the threshold invariant to vocabulary size and top-k across models. We adopt $H_b(0.99) \approx 0.0561$ nats, and an ablation over $p_1 \in \{0.90, 0.95, 0.99, 0.999\}$ confirms that our findings in Section 5.2 are consistent across these different thresholds (see Appendix J).

Using this threshold, we introduce a *cliff taxonomy* with three types:

Table 1: Distribution of cliff types: deterministic (D), uncertain (U), and sampled-off (S). Cliff and all-token proportions give each type's percentage among identified cliff tokens and all reasoning-trace tokens, respectively. Darker shading indicates larger values.

	Cliff proportion (%)			All-token proportion (%)		
Model	D	U	S	D	U	S
`Qwen3-8B`	47.9	34.4	17.7	69.1	27.8	3.1
`Qwen3-4B`	44.4	36.1	19.4	73.9	23.6	2.5
`Qwen3-0.6B`	29.8	41.9	28.2	64.3	32.2	3.5
`Llama-3.1-8B`	22.0	25.2	52.8	66.8	28.3	4.8
`Llama-3.2-3B`	16.8	25.3	54.7	73.5	22.4	4.0
`Llama-3.2-1B`	9.0	28.8	62.2	74.3	22.0	3.6
`Gemma-3-4B`	45.2	29.8	21.4	77.6	19.7	2.7

(1) **Deterministic cliff**: a greedy token with $H_t < 0.0561$; the model samples the cliff token with near certainty.
(2) **Uncertain cliff**: a greedy token with $H_t \geq 0.0561$; the greedy cliff token is sampled despite high uncertainty.
(3) **Sampled-off cliff**: a non-greedy token with $H_t \geq 0.0561$; a non-greedy cliff token is stochastically sampled under high uncertainty.

The fourth type (non-greedy token, $H_t < 0.0561$) is excluded from our analysis due to its extreme rarity by definition; as it requires sampling a token with probability $p \leq 0.01$, only six cases were observed (three within `Gemma-3-4B` and three within `Llama-3.2-3B`).

5.2 Cliff taxonomy varies by model family and scale

We first compare the cliff-token distributions against the all-token distributions. As shown in Table 1, deterministic cliffs account for a smaller proportion of cliff tokens compared to their all-token proportion, whereas sampled-off cliffs occur at a higher proportion across all models. This indicates that cliff tokens are not randomly distributed errors. Instead, they show a distinct distribution pattern from normal tokens.

Furthermore, analyzing models within the same family shows a consistent scaling effect. As model size increases in the `Qwen3` and `Llama-3` families, the proportion of sampled-off cliffs decreases, while the proportion of deterministic cliffs increases. This trend suggests that scaling up model parameters shifts the errors from stochastic sampling to deterministic ones. Beyond scale, the dominant cliff type is influenced by the model's family. Similar-sized models, such as `Qwen3-8B` and `Llama-3.1-8B`, exhibit different cliff-token distributions. `Qwen3-8B` produces more deterministic cliffs, whereas `Llama-3.1-8B` produces more sampled-off cliffs. This shows that each model family has a specific pattern in its reasoning failures. Overall, these findings show that token-level mathematical reasoning failures vary with both model family and scale. This heterogeneity has direct implications for cliff-targeted post-training, which we explore in Section 6.

6 Do cliff types differ as training signals?

While prior work has demonstrated the effectiveness of post-training with token-level supervision (Lin et al., 2025; Abdin et al., 2024), these approaches treat token-level signals as a homogeneous supervision signal. This raises the question: Do the three cliff types differ in their effectiveness as training signals?

6.1 Methodology

Cliff-DPO loss. Given the cliff preference dataset $\mathcal{D}_{\text{cliff}} = \{(\boldsymbol{x}_i, \boldsymbol{c}_{i,<t_i}, c^w_{i,t_i}, c^l_{i,t_i})\}_{i=1}^M$, we adapt the standard sigmoid Direct Preference Optimization (DPO) loss (Rafailov et al., 2023) to provide single-token supervision specifically at each identified cliff position t_i. We refer to this objective as

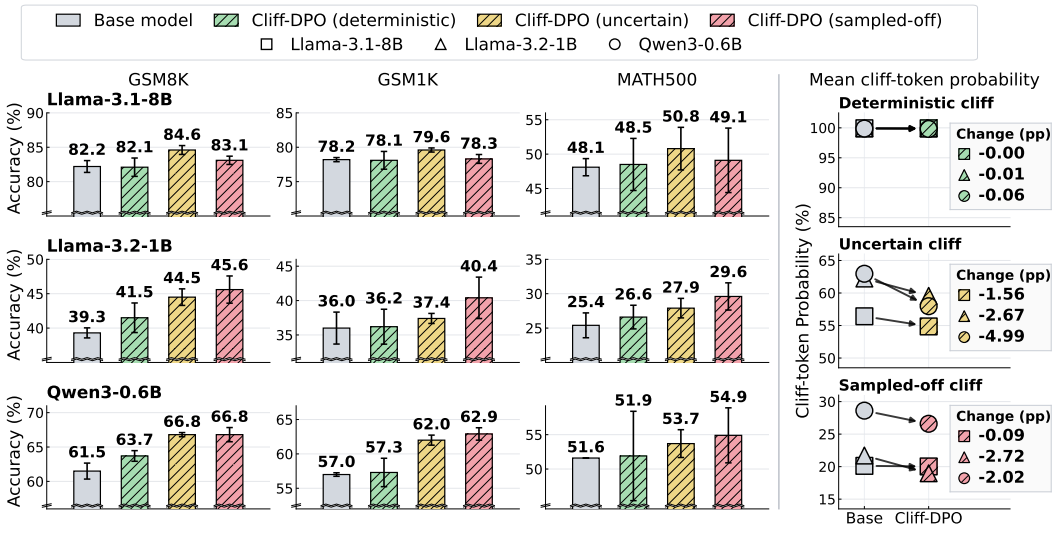


Figure 5: **(Left)** Cliff-DPO performance across models and benchmarks. Mean accuracy and 95% confidence intervals are reported. **(Right)** Mean cliff-token probability before and after Cliff-DPO on held-out cliff positions from GSM1K, MATH500, and AIME 2025. Each variant is evaluated on its corresponding cliff type; changes are in percentage points (pp). Full results are in Appendix O.

Cliff-DPO. The loss is formulated as follows:

$$\mathcal{L}_{\text{Cliff-DPO}}(\theta) = -\frac{1}{M}\sum_{i=1}^{M} \log \sigma\left(r_\theta(c^w_{i,t_i}; \boldsymbol{x}_i, \boldsymbol{c}_{i,<t_i}) - r_\theta(c^l_{i,t_i}; \boldsymbol{x}_i, \boldsymbol{c}_{i,<t_i})\right), \tag{6}$$

where r_θ denotes the implicit pointwise reward, defined as

$$r_\theta(c_t; \boldsymbol{x}, \boldsymbol{c}_{<t}) := \beta \log \frac{\pi_\theta(c_t \mid \boldsymbol{x}, \boldsymbol{c}_{<t})}{\pi_{\text{ref}}(c_t \mid \boldsymbol{x}, \boldsymbol{c}_{<t})}. \tag{7}$$

Here, c^w_{i,t_i} and c^l_{i,t_i} are the non-cliff token (chosen) and cliff token (rejected) at position t_i, respectively. The coefficient β controls the scale of the implicit reward and the strength of reference-policy regularization. By applying the preference loss only to the candidate tokens at the cliff position, Cliff-DPO localizes the training signal to the point where the reasoning trace diverges into failure.

Constructing Cliff-DPO training pairs. To construct $\mathcal{D}_{\text{cliff}}$, we apply the pipeline in Section 3.3 to the GSM8K (Cobbe et al., 2021) training set of 7,473 problems and identify cliff positions. At each cliff position t, we consider the top-10 candidate tokens under the model distribution and perform $N = 64$ rollouts for each candidate to estimate its success probability. We define $C_t^{\text{non-cliff}}$ as the set of candidates whose estimated success probability does not indicate a cliff token. Each non-cliff token $c_t^w \in C_t^{\text{non-cliff}}$ is then paired with the originally detected cliff token c_t^l, yielding a chosen–rejected preference pair. These preference pairs are then divided into three subsets according to the type of the rejected cliff token: *deterministic*, *uncertain*, or *sampled-off*.

Experimental setup. As identifying cliff tokens across 7,473 problems requires extensive rollouts at scale, we use three models of different sizes and families: `Llama-3.1-8B`, `Llama-3.2-1B`, and `Qwen3-0.6B`. We train them across three distinct data configurations to analyze the impact of different cliff types: *deterministic*, *uncertain*, and *sampled-off*.[3] See Appendix Q for training and evaluation details and Appendix N for results on additional Cliff-DPO variants and comparisons with three baselines: the base model, DPO (Rafailov et al., 2023), and cDPO (Lin et al., 2025).

[3]To control for the differing pair counts across the cliff types, we also train size-matched variants. See Appendix P for further details and results.

6.2 Results and discussion

Figure 5 shows that both the *uncertain* variant and the *sampled-off* variant achieve higher mean accuracy than the base model and the *deterministic* variant on GSM8K, GSM1K, and MATH500 across all three models. The best-performing variant differs by model: on average across these benchmarks, the *sampled-off* variant ranks highest for `Qwen3-0.6B` and `Llama-3.2-1B`, whereas the *uncertain* variant ranks highest for `Llama-3.1-8B`. The *deterministic* variant shows smaller gains for the two smaller models and leaves `Llama-3.1-8B` performance largely unchanged on these three benchmarks. Overall, the *uncertain* variant and the *sampled-off* variant are more effective in the evaluated settings on the three benchmarks.

The effect of Cliff-DPO depends on how strongly the model initially favors the cliff token. For deterministic cliffs, the token probability is more than 99%, indicating that the prior is difficult to overcome via post-training. Accordingly, Cliff-DPO changes its mean cliff-token probability by at most 0.06 pp. For uncertain and sampled-off cliffs, the model assigns lower probability to the cliff token, giving post-training more room to reduce its token probability. Cliff-DPO therefore produces larger reductions of token probability for uncertain cliffs (up to 4.99 pp) and sampled-off cliffs (up to 2.72 pp).

7 Conclusion

We introduce **cliff tokens**, single tokens where the success probability of a reasoning trace collapses. We formalize cliff tokens using a z-test-based adaptive threshold, separating statistically significant drops from sampling noise. Across evaluated model–benchmark settings where cliff tokens are observed, deleting the first cliff token generally outperforms retaining it, showing that cliff tokens act as failure triggers. Starting with a clear gap at $k = 1$, this advantage persists across larger k, reaching a pass@64 of 1.0, whereas retaining the token yields 0.86 on average. We also find two distinct properties of cliff tokens: they do not necessarily coincide with explicit error tokens, and they do not guarantee failure, since the model can recover through self-correction. We further introduce a cliff taxonomy—deterministic, uncertain, and sampled-off cliffs—and show that its distribution varies by model family and scale. Finally, using Cliff-DPO, we show that single-token supervision at cliff positions improves mean accuracy, with the *uncertain* and *sampled-off* variants showing larger gains than the *deterministic* variant across three models on GSM8K, GSM1K, and MATH500. Overall, cliff tokens offer a token-level view for diagnosing reasoning failures and a targeted training signal for improving mathematical reasoning in LLMs.

Limitations and future work. Our framework characterizes failure triggers, but its scope is limited to cases where the model can reach the correct answer but loses that chance during reasoning. When a problem is beyond the model's capacity, success probability is near zero from the start, so no cliff token appears. In 30 incorrect `Llama-3.2-1B` traces on AIME 2025, we observed no cliff tokens because the success probability remained below 0.05 throughout all 30 traces. Our experiments are constrained by the computational cost of success probability estimation, which required $4{,}047$ A100 (80GB) GPU-hours across seven models and three benchmarks (Appendix C). Due to the substantial compute requirements, we subsampled GSM1K and MATH500 rather than evaluating their full sets. Although Cliff-DPO uses few loss-contributing tokens (Appendix N), obtaining cliff tokens still requires costly rollout-based detection. Future work should develop efficient predictors, which could enable decoding-time resampling, backtracking, or branching before a reasoning trace collapses. Finally, our analysis is limited to mathematical reasoning. Whether these findings generalize to larger models, longer reasoning traces, and non-mathematical domains remains an open question.

AI use statement

In this work, we used generative AI tools to assist in implementing the cliff detection, token-level rollout, and Cliff-DPO pipelines, checking sampling and decoding settings before running experiments, and writing scripts that compute the statistics reported in our tables from the raw rollout and evaluation outputs. We did not use them to form or refine hypotheses, to define cliff tokens or the cliff taxonomy, or to interpret results. Synthetic data generation and mathematical proofs are not applicable to this work. We also used AI tools to find and summarize related work, to write plotting

code for figures we designed, to format tables and other content in LaTeX, and to polish the writing. We have reviewed all AI-assisted work. The authors ran, inspected, and tested all AI-assisted code, and checked every reported number and plotted value against the original outputs. We read every cited paper ourselves and checked AI-generated summaries against the originals. We also checked all AI-edited text for accuracy and consistency with our experiments. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Ethics statement

This work does not involve experiments with human participants or any private data. All experiments use publicly available math benchmarks (GSM1K, GSM8K, MATH500, and AIME 2025) and open-weight models, following their licenses and terms of use. We release the model-generated reasoning traces and token-level rollout data used in our analysis. The released data contain only benchmark problems and model outputs, and remain subject to the original dataset licenses and model terms of use. This work analyzes where language models fail at mathematical reasoning, and we do not expect it to enable harmful applications. Although Cliff-DPO improves accuracy on math benchmarks, these gains may not generalize to other domains, so applying our methods in high-stakes settings would require further validation and human oversight.

Reproducibility statement

We describe the cliff-token definition and the adaptive threshold in Section 3.1, and the models, datasets, base threshold, and rollout protocol in Sections 3.2 and 3.3. Appendix D gives the dataset subsampling procedure and random seed, the sampling hyperparameters for each model, the maximum generation lengths, and the prompt template. The Cliff-del and Cliff-keep setups are defined in Section 4.2, and the bootstrap procedure for their confidence intervals is described in Appendix G. The entropy threshold for the cliff taxonomy is defined in Section 5.1, with sensitivity analyses of the base thresholds and entropy thresholds in Appendices A and J. For Cliff-DPO, Section 6.1 describes the loss and the construction of preference pairs, and Appendix Q lists the training and evaluation configurations for Cliff-DPO and both baselines, including hyperparameters, LoRA settings, and seeds. Appendix P describes the token-matched ablation, Appendix C reports the computational cost, and Appendix R lists the licenses of all models and datasets. Our anonymous repository (`https://anonymous.4open.science/r/cliff_token_data/`) contains the full source code and scripts for each experiment. It also includes the data used in this work: the stem traces and token-level success probabilities from 64 rollouts for all seven models, together with the GSM8K training traces used to build the Cliff-DPO pairs, so that our analyses can be reproduced without regenerating the costly rollouts.

References

Marah I Abdin, Jyoti Aneja, Harkirat S. Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

Gregor Bachmann, Yichen Jiang, Seyed-Mohsen Moosavi-Dezfooli, and Moin Nabi. The potential of CoT for reasoning: A closer look at trace dynamics. In *ICLR*, 2026.

Eric Bigelow, Amir Zur, Satchel Grant, Tal Haklay, Can Rager, Owen Lewis, Thomas McGrath, Jack Merullo, Ekdeep Singh Lubana, and Atticus Geiger. Forking fast: Efficiently estimating uncertainty dynamics in text generation. *arXiv preprint arXiv:2608.19611*, 2026.

Eric J. Bigelow, Ari Holtzman, Hidenori Tanaka, and Tomer D. Ullman. Forking paths in neural text generation. In *ICLR*, 2025.

Paul C. Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. Thought anchors: Which LLM reasoning steps matter? *arXiv preprint arXiv:2506.19143*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean-bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, and 196 others. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 538 others. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 175 others. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645, 2025.

Hasan Abed Al Kader Hammoud, Hani Itani, and Bernard Ghanem. Beyond the last answer: Your reasoning trace uncovers more than you think. *arXiv preprint arXiv:2504.20708*, 2025.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, Kai Dang, Yang Fan, Yichang Zhang, An Yang, Rui Men, Fei Huang, Bo Zheng, Yibo Miao, Shanghaoran Quan, and 5 others. Qwen2.5-Coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Hyeonbin Hwang, Doyoung Kim, Seungone Kim, Seonghyeon Ye, and Minjoon Seo. Self-Explore: Enhancing mathematical reasoning in language models with fine-grained rewards. In *Findings of EMNLP*, 2024.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, and 11 others. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *ICLR*, 2024.

Zicheng Lin, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xing Wang, Ruilin Luo, Chufan Shi, Siheng Li, Yujiu Yang, and Zhaopeng Tu. Critical tokens matter: Token-level contrastive estimation enhances LLM's reasoning capability. In *ICML*, 2025.

Aiwei Liu, Haoping Bai, Zhiyun Lu, Yanchao Sun, Xiang Kong, Xiaoming Simon Wang, Jiulong Shan, Albin Madappally Jose, Xiaojiang Liu, Lijie Wen, Philip S. Yu, and Meng Cao. TIS-DPO: Token-level importance sampling for direct preference optimization with estimated weights. In *ICLR*, 2025a.

Junnan Liu, Hongwei Liu, Linchen Xiao, Ziyi Wang, Kuikun Liu, Songyang Gao, Wenwei Zhang, Songyang Zhang, and Kai Chen. Are your LLMs capable of stable reasoning? In *Findings of ACL*, 2025b.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Meiqi Guo, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, and Abhinav Rastogi. Improve mathematical reasoning in language models by automated process supervision. *arXiv preprint arXiv:2406.06592*, 2024.

OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 243 others. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

OpenCompass. AIME 2025. `https://huggingface.co/datasets/opencompass/AIME2025`, 2025. Hugging Face dataset.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Sam Silver, Jimin Sun, Ivan Zhang, Sara Hooker, and Eddie Kim. Language models can perform single-utterance self-correction of perturbed reasoning. *arXiv preprint arXiv:2506.15894*, 2025.

Zetian Sun, Dongfang Li, Baotian Hu, and Min Zhang. Improving value-based process verifier via low-cost variance reduction. In *AAAI*, 2026.

Jean-Francois Ton, Muhammad Faaiz Taufiq, and Yang Liu. Understanding chain-of-thought in LLMs through information theory. In *ICML*, 2025.

Alexander von Recum, Leander Girrbach, and Zeynep Akata. Are reasoning LLMs robust to interventions on their chain-of-thought? In *ICLR*, 2026.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *ACL*, 2024.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In *NeurIPS*, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

Shangyu Xing, Siyuan Wang, Chenyuan Yang, Xinyu Dai, and Xiang Ren. Lookahead tree-based rollouts for enhanced trajectory-level exploration in reinforcement learning with verifiable rewards. In *ICLR*, 2026.

Wei Xiong, Wenting Zhao, Weizhe Yuan, Olga Golovneva, Tong Zhang, Jason Weston, and Sainbayar Sukhbaatar. StepWiser: Stepwise generative judges for wiser reasoning. *arXiv preprint arXiv:2508.19229*, 2025.

Haobo Xu, Sirui Chen, Ruizhong Qiu, Yuchen Yan, Chen Luo, Monica Xiao Cheng, Jingrui He, and Hanghang Tong. Prune as you generate: Online rollout pruning for faster and better RLVR. In *ACL*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yongcheng Zeng, Guoqing Liu, Weiyu Ma, Ning Yang, Haifeng Zhang, and Jun Wang. Token-level direct preference optimization. In *ICML*, 2024.

Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Charlotte Zhuang, Dylan Slack, Qin Lyu, Sean Hendryx, Russell Kaplan, Michele Lunati, and Summer Yue. A careful examination of large language model performance on grade school arithmetic. In *NeurIPS*, 2024.

Kechi Zhang, Ge Li, Jia Li, Yihong Dong, Jia Li, and Zhi Jin. Focused-DPO: Enhancing code generation through focused preference optimization on error-prone points. In *Findings of ACL*, 2025.

Yufeng Zhang, Xuepeng Wang, Lingxiang Wu, and Jinqiao Wang. Characterizing and mitigating reasoning drift in large language models. In *ICLR*, 2026.

Mingkang Zhu, Xi Chen, Zhongdao Wang, Bei Yu, Hengshuang Zhao, and Jiaya Jia. TGDPO: Harnessing token-level reward guidance for enhancing direct preference optimization. In *ICML*, 2025.

Wei Zhu, Jian Zhang, Lixing Yu, Kun Yue, and Zhiwen Tang. Dissecting failure dynamics in large language model reasoning. In *ACL*, 2026.

A SENSITIVITY ANALYSIS OF THE ADAPTIVE THRESHOLD

Because SE_t in the adaptive threshold depends on the estimated success probabilities before and after the token, the resulting adaptive threshold varies across possible pairs (p_{t-1}, p_t). We conduct a sensitivity analysis by varying the base threshold from 0.1 to 0.4.

In Table 2, the *Adaptive threshold (min/max)* columns report the lower and upper bounds of this adaptive threshold for each base threshold. Detection counts are aggregated over GSM1K, MATH500, and AIME 2025 for each model. This analysis evaluates how changing the base threshold affects the number of detected cliff tokens across models and datasets.

Table 2: Sensitivity of cliff-token counts to the base threshold. The *Total* column sums counts across the seven evaluated models.

	Adaptive threshold		**# Cliff tokens detected**							
Baseline threshold	Min	Max	Qwen3 -8B	Qwen3 -4B	Qwen3 -0.6B	Llama -3.1-8B	Llama -3.2-3B	Llama -3.2-1B	Gemma -3-4B	**Total**
0.1	0.180	0.241	96	72	124	127	95	111	84	709
0.2	0.294	0.337	29	14	24	43	40	41	11	202
0.3	0.401	0.431	17	6	15	30	21	24	7	120
0.4	0.503	0.523	11	2	10	22	16	15	4	80

B GRANULARITY OF CLIFF-TOKEN DETECTION

To obtain cliff tokens, we performed rollouts from *every* token position. While this approach precisely localizes the cliff token, it is computationally expensive. In this section, we investigate whether rollouts can instead be performed every c-th token ($c > 1$).

We evaluate fixed-size non-overlapping chunk detection on the same seven models and three benchmarks analyzed throughout this work, comprising $1{,}610$ traces and 709 cliff tokens, using chunk sizes of 4, 8, 16, and 32 tokens. A chunk is flagged when the drop in success probability between its boundaries exceeds the adaptive threshold (Section 3.1), and a flagged chunk counts as a hit when it contains a cliff token. By definition, a chunk size of 1 corresponds to the original cliff-token detector, yielding perfect precision, recall, and F1 scores of 100%.

As reported in Table 3, chunk-level detection misses the majority of cliff tokens, yielding persistently low recall across all chunk sizes. Reducing the chunk size does not resolve this issue. This performance limitation stems from the non-monotonicity of token-wise success probabilities (Bachmann et al., 2026). Because chunk-level evaluation only compares the boundary values of a chunk, it cannot capture internal token-level variations of success probability. First, a sharp drop can be masked by a subsequent recovery in success probability inside the same chunk. Second, the accumulation of minor drops across a chunk can create a large boundary difference, leading to false positive detections. Consequently, chunk-level detection cannot provide the precise localization required for our analysis.

Table 3: Chunk-level detection of cliff tokens.

Chunk size	**Precision**	**Recall**	**F1**
4	60.2%	40.7%	48.6%
8	60.8%	37.9%	46.7%
16	62.1%	35.9%	45.5%
32	62.3%	34.5%	44.4%

C COMPUTATIONAL COST OF SUCCESS PROBABILITY ESTIMATION

C.1 COMPUTATIONAL COMPLEXITY

The computational load required to obtain the success probability of a given reasoning trace scales significantly with both the number of rollouts N and the granularity of the estimation steps. Our success probability analysis demands estimation at every single token position on the reasoning trace to accurately capture sudden reasoning drifts (i.e., cliff tokens).

Suppose a generated reasoning trace consists of a total of T tokens. At each token position $t \in \{1, \ldots, T\}$, estimating the success probability requires performing N rollouts, generating the remaining sequence to reach the final outcome. Assuming the average total length of a complete reasoning trace remains approximately T, the number of tokens generated for the rollouts at position t is $N(T - t)$.

Therefore, the total number of tokens T_{tot} produced to estimate the full success probability curve from scratch for a single reasoning trace is given by:

$$T_{\text{tot}} = \sum_{t=1}^{T} N(T - t) = N\frac{T(T-1)}{2} \approx \frac{NT^2}{2} \tag{8}$$

This demonstrates a quadratic scaling $\mathcal{O}(NT^2)$ with respect to the reasoning length T. In the context of mathematical reasoning, where T is inherently large due to the multi-step nature of the CoT, this quadratic complexity imposes a severe computational bottleneck. This fundamental constraint justifies our methodological choice in Section 3: rather than attempting an intractable scale-up of N, we introduce a z-test to establish the statistical significance of Cliff tokens under a feasible computational budget.

C.2 EMPIRICAL GPU COST

In total, success probability estimation required approximately $4{,}047$ A100 (80GB) GPU-hours across seven models and three benchmarks. This cost is dominated by rollout generation and varies substantially across benchmarks because the number of rollout tokens grows with the length of the original reasoning trace. As discussed in Appendix C.1, estimating success probability scales quadratically with the reasoning length, so benchmarks with longer reasoning traces incur much larger computational cost.

All rollout jobs were run with data parallelism over 8 A100 (80GB) GPUs. Under this setup, success probability estimation consumed approximately 392 GPU-hours for GSM1K (700 traces), $2{,}087$ GPU-hours for MATH500 (700 traces), and $1{,}568$ GPU-hours for AIME 2025 (210 traces). The large difference across benchmarks reflects both the number of traces and the length of the generated reasoning traces. In particular, although AIME 2025 has fewer traces, its per-trace cost is substantially higher than GSM1K because AIME 2025 traces are much longer and require many more rollout tokens for success probability estimation.

D EXPERIMENTAL DETAILS

D.1 DATASET PREPROCESSING

For analyzing GSM1K and MATH500, we randomly sampled 100 problems each using a fixed random seed of 42, while including all 30 problems from AIME 2025.

D.2 SAMPLING HYPERPARAMETERS

For both generating the initial reasoning traces and performing the subsequent rollouts for success probability estimation, we utilize the hyperparameters detailed in Table 4. These settings follow the official default configurations recommended for each respective model to ensure optimal reasoning performance.

Table 4: Sampling hyperparameters used for all experiments. "–" in the top-k column denotes that top-k truncation is disabled.

Model	Temperature	Top-p	Top-k
`Qwen3-8B`	0.7	0.80	20
`Qwen3-4B`	0.7	0.80	20
`Qwen3-0.6B`	0.7	0.80	20
`Llama-3.1-8B`	0.6	0.90	–
`Llama-3.2-3B`	0.6	0.90	–
`Llama-3.2-1B`	0.6	0.90	–
`Gemma-3-4B`	1.0	0.95	64

D.3 MAXIMUM TOKEN LENGTHS

As mentioned in Section 3.3, to maintain computational efficiency while accommodating the varying reasoning complexities of the datasets, we set dataset-specific maximum token limits. As detailed in Table 5, the maximum generation lengths during rollouts are strictly bounded to $1{,}024$ tokens for GSM1K/GSM8K, and $2{,}048$ tokens for MATH500. For AIME 2025, the rollout budget is specifically restricted to $4{,}096$ tokens to prevent models generating overly long traces that make rollout sampling infeasible, while allowing a reasonable budget for solving the challenging problems.

Table 5: Maximum generation tokens per dataset. *Stem traces* denotes the budget for generating the initial stem trace per problem. *Rollouts* indicates the budget during success probability estimation. *Evaluation* represents the extended budget used when evaluating the final accuracy of each Cliff-DPO model in Section 6.

Dataset	Stem traces	Rollouts	Evaluation
GSM8K	1,024	1,024	8,192
GSM1K	1,024	1,024	8,192
MATH500	2,048	2,048	16,384
AIME 2025	8,192	4,096	32,768

D.4 PROMPT TEMPLATE

For all seven models across all datasets, we employ the same zero-shot prompt. We append the following instruction to the end of every input query:

Prompt. *Please reason step by step, and put your final answer within* `\boxed{}`.

E CLIFF TOKENS VS. CRITICAL TOKENS

Comparison setup. We compare cliff tokens with critical tokens from Lin et al. (2025). In their definition, a critical token is a token whose success probability is 0 and whose success probabilities for all subsequent tokens remain below 0.05. In contrast, a cliff token is a token at which the drop in success probability exceeds our z-test-based adaptive threshold with $\tau = 0.1$. We evaluate both notions on the same incorrect subset of traces using deletion-based recovery:

- **Cliff-del:** resampling from the prefix before the first cliff token.
- **Critical-del:** resampling from the prefix before the critical token.

For each intervention, avg@64 is computed as the fraction of 64 temperature-sampled rollouts that yield a correct answer, and each table entry reports the mean avg@64 across traces in the corresponding incorrect subset. If an incorrect trace does not contain the corresponding target token, we assign avg@64 $= 0$ for that intervention. See Appendix D for experimental details.

Results. Table 6 shows that Cliff-del outperforms Critical-del in 18 of the 19 comparable model–dataset settings. Two additional settings, `Llama-3.1-8B` on AIME 2025 and `Llama-3.2-1B` on AIME 2025, are not directly comparable because no cliff token was identified in any incorrect trace. The only comparable exception is `Llama-3.2-3B` on AIME 2025, where the avg@64 under both interventions are at most 0.011, indicating that neither deletion substantially restores correctness.

Overall, deleting the first cliff token leads to higher recovery than deleting the critical token in the majority of comparable settings. This supports the interpretation that cliff tokens identify the local point where the model shifts into an incorrect trace. Thus, the two notions capture different stages of failure: critical tokens correspond to points where an error has become persistent, whereas cliff tokens identify an earlier trigger associated with the collapse of the reasoning trace.

Table 6: Mean avg@64 after Cliff-del vs. Critical-del on the incorrect subset. **Bold** marks the larger value within each (model, dataset) block. '-' indicates no cliff token was identified in any incorrect trace for that model and dataset pair.

	GSM1K		**MATH500**		**AIME 2025**	
Model	Cliff-del	Critical-del	Cliff-del	Critical-del	Cliff-del	Critical-del
`Qwen3-8B`	**0.161**	0.073	**0.284**	0.078	**0.037**	0.029
`Qwen3-4B`	**0.132**	0.047	**0.312**	0.101	**0.072**	0.021
`Qwen3-0.6B`	**0.238**	0.106	**0.187**	0.084	**0.012**	0.002
`Llama-3.1-8B`	**0.434**	0.115	**0.267**	0.098	-	0.008
`Llama-3.2-3B`	**0.336**	0.149	**0.173**	0.098	0.006	**0.011**
`Llama-3.2-1B`	**0.230**	0.090	**0.125**	0.051	-	0.001
`Gemma-3-4B`	**0.308**	0.076	**0.239**	0.054	**0.026**	0.010

F CLIFF-TOKEN MULTIPLICITY AND DOWNSTREAM CORRECTNESS

To complement the analysis in Section 4.3, we report model-wise evidence that cliff tokens occur within non-monotonic reasoning dynamics, where the estimated success probability fluctuates along the trace (Bachmann et al., 2026). Based on Table 7, we identify two primary patterns in traces containing at least one cliff token. First, nearly half of these traces (47.2%) contain multiple cliff tokens, indicating that reasoning errors frequently occur as repeated local failures. Second, a cliff token does not strictly determine an incorrect outcome; approximately one-third of the traces with cliff tokens (32.4%) successfully recover to reach the correct final answer.

The proportion of traces that still reach the correct answer after cliff tokens increases with model size in both the `Qwen3` and `Llama-3` families. This pattern is consistent with greater downstream recovery capacity in larger models (von Recum et al., 2026; Silver et al., 2025), potentially reflecting stronger self-correction.

Table 7: Multiplicity and downstream correctness among traces containing at least one cliff token. Percentages are relative to the number of traces with any cliff token. Multiple-cliff traces contain at least two cliff tokens, and correct traces end with the correct answer.

Model	**Traces with any cliff token**	**Multiple-cliff traces, n (%)**	**Correct traces, n (%)**
`Qwen3-8B`	33	14 (42.4%)	17 (51.5%)
`Qwen3-4B`	29	20 (69.0%)	12 (41.4%)
`Qwen3-0.6B`	58	36 (62.1%)	16 (27.6%)
`Llama-3.1-8B`	74	29 (39.2%)	26 (35.1%)
`Llama-3.2-3B`	56	25 (44.6%)	14 (25.0%)
`Llama-3.2-1B`	69	25 (36.2%)	16 (23.2%)
`Gemma-3-4B`	39	20 (51.3%)	15 (38.5%)
Total	**358**	**169 (47.2%)**	**116 (32.4%)**

G DETAILED CLIFF-DEL AND CLIFF-KEEP RESULTS

This section reports the complete pass@k results for the Cliff-del and Cliff-keep interventions described in Section 4.2. The tables provide mean pass@k values, pointwise 95% two-level bootstrap confidence intervals (CIs), paired differences between the interventions (Δ), and the number of traces included in the analysis for each dataset–model combination. We repeat the following procedure $10{,}000$ times to calculate CIs: At each bootstrap iteration, we first resample traces with replacement and then resample the 64 rollout outcomes within each selected trace.

G.1 RESULTS FOR INCORRECT TRACES

In Section 4.2, the main analysis focuses on incorrect traces, where deleting the first cliff token recovers reasoning performance. Tables 8 to 10 show detailed results for GSM1K, MATH500, and AIME 2025, respectively. Cliff-del yields equal or higher mean pass@k compared to Cliff-keep in most settings, with a few exceptions. Cliff-del reaches pass@64 of 1.0 in every model and benchmark setting included in the analysis, while Cliff-keep ranges from 0.67 to 1.0.

Table 8: **GSM1K:** Cliff-del and Cliff-keep pass@k on incorrect traces. Cells report mean pass@k across traces and pointwise 95% CIs. Δ is Cliff-del minus Cliff-keep and uses paired trace resamples. n is the number of traces that contain at least one cliff token.

Model	Setup	$k=1$	$k=2$	$k=4$	$k=8$	$k=16$	$k=32$	$k=64$
`Qwen3-8B` ($n=4$)	Cliff-del	0.43 [0.34,0.52]	0.68 [0.56,0.77]	0.89 [0.79,0.95]	0.99 [0.95,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.28 [0.18,0.38]	0.48 [0.33,0.61]	0.72 [0.54,0.85]	0.91 [0.77,0.98]	0.99 [0.92,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]
	Δ	0.15 [-0.01,0.31]	0.20 [-0.01,0.41]	0.17 [-0.02,0.38]	0.08 [-0.01,0.22]	0.01 [0.00,0.08]	0.00 [0.00,0.01]	0.00 [0.00,0.00]
`Qwen3-4B` ($n=4$)	Cliff-del	0.38 [0.22,0.53]	0.60 [0.38,0.77]	0.81 [0.59,0.95]	0.94 [0.78,1.00]	0.99 [0.92,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.36 [0.23,0.47]	0.58 [0.39,0.72]	0.81 [0.61,0.92]	0.94 [0.81,1.00]	0.99 [0.94,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]
	Δ	0.02 [-0.08,0.13]	0.02 [-0.11,0.15]	0.00 [-0.11,0.11]	-0.01 [-0.10,0.07]	0.00 [-0.06,0.04]	0.00 [-0.01,0.00]	0.00 [0.00,0.00]
`Qwen3-0.6B` ($n=20$)	Cliff-del	0.63 [0.54,0.71]	0.83 [0.76,0.88]	0.96 [0.92,0.98]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.38 [0.29,0.47]	0.57 [0.47,0.67]	0.77 [0.68,0.85]	0.93 [0.87,0.96]	0.99 [0.97,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.25 [0.17,0.34]	0.25 [0.17,0.34]	0.18 [0.11,0.27]	0.07 [0.04,0.13]	0.01 [0.00,0.03]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Llama-3.1-8B` ($n=21$)	Cliff-del	0.60 [0.47,0.72]	0.76 [0.64,0.87]	0.88 [0.79,0.95]	0.96 [0.91,0.99]	0.99 [0.97,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.23 [0.15,0.33]	0.37 [0.24,0.50]	0.52 [0.35,0.67]	0.63 [0.45,0.79]	0.72 [0.53,0.86]	0.79 [0.59,0.92]	0.86 [0.62,0.95]
	Δ	0.37 [0.23,0.50]	0.39 [0.25,0.54]	0.37 [0.21,0.53]	0.32 [0.16,0.50]	0.27 [0.12,0.46]	0.21 [0.08,0.41]	0.14 [0.05,0.38]
`Llama-3.2-3B` ($n=20$)	Cliff-del	0.59 [0.48,0.70]	0.78 [0.68,0.86]	0.92 [0.85,0.96]	0.98 [0.95,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.32 [0.20,0.44]	0.46 [0.31,0.61]	0.59 [0.42,0.74]	0.70 [0.53,0.84]	0.81 [0.64,0.91]	0.90 [0.74,0.97]	0.95 [0.80,1.00]
	Δ	0.27 [0.15,0.41]	0.32 [0.18,0.46]	0.33 [0.18,0.48]	0.28 [0.15,0.44]	0.19 [0.08,0.35]	0.10 [0.03,0.26]	0.05 [0.00,0.20]
`Llama-3.2-1B` ($n=31$)	Cliff-del	0.52 [0.43,0.61]	0.71 [0.63,0.79]	0.87 [0.81,0.92]	0.97 [0.93,0.98]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.20 [0.12,0.28]	0.31 [0.20,0.42]	0.44 [0.30,0.56]	0.55 [0.40,0.68]	0.66 [0.49,0.78]	0.74 [0.56,0.85]	0.81 [0.61,0.90]
	Δ	0.32 [0.23,0.42]	0.40 [0.29,0.51]	0.44 [0.31,0.56]	0.41 [0.28,0.56]	0.34 [0.22,0.50]	0.26 [0.15,0.44]	0.19 [0.10,0.39]
`Gemma-3-4B` ($n=8$)	Cliff-del	0.67 [0.56,0.77]	0.87 [0.79,0.94]	0.98 [0.94,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.49 [0.26,0.67]	0.66 [0.35,0.89]	0.74 [0.40,0.99]	0.77 [0.44,1.00]	0.78 [0.49,1.00]	0.81 [0.50,1.00]	0.88 [0.50,1.00]
	Δ	0.18 [0.02,0.37]	0.22 [0.01,0.47]	0.23 [0.00,0.53]	0.23 [0.00,0.55]	0.22 [0.00,0.50]	0.19 [0.00,0.50]	0.13 [0.00,0.50]

Table 9: **MATH500:** Cliff-del and Cliff-keep pass@k on incorrect traces. Notations follow Table 8.

Model	Setup	$k=1$	$k=2$	$k=4$	$k=8$	$k=16$	$k=32$	$k=64$
`Qwen3-8B` ($n=9$)	Cliff-del	0.45 [0.37,0.54]	0.69 [0.59,0.78]	0.89 [0.80,0.95]	0.98 [0.94,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.34 [0.19,0.48]	0.52 [0.31,0.70]	0.68 [0.43,0.88]	0.78 [0.53,0.97]	0.84 [0.58,1.00]	0.88 [0.64,1.00]	0.89 [0.67,1.00]
	Δ	0.12 [-0.01,0.27]	0.17 [0.00,0.38]	0.21 [0.02,0.44]	0.20 [0.01,0.45]	0.16 [0.00,0.41]	0.12 [0.00,0.36]	0.11 [0.00,0.33]
`Qwen3-4B` ($n=9$)	Cliff-del	0.58 [0.48,0.68]	0.81 [0.71,0.89]	0.95 [0.89,0.98]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.43 [0.30,0.54]	0.64 [0.47,0.78]	0.82 [0.64,0.94]	0.92 [0.78,1.00]	0.97 [0.88,1.00]	1.00 [0.94,1.00]	1.00 [1.00,1.00]
	Δ	0.15 [0.06,0.25]	0.16 [0.06,0.28]	0.13 [0.03,0.27]	0.07 [0.00,0.21]	0.03 [0.00,0.12]	0.00 [0.00,0.06]	0.00 [0.00,0.00]
`Qwen3-0.6B` ($n=21$)	Cliff-del	0.44 [0.35,0.54]	0.64 [0.54,0.74]	0.82 [0.72,0.90]	0.93 [0.85,0.98]	0.98 [0.92,1.00]	1.00 [0.96,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.31 [0.20,0.42]	0.46 [0.31,0.60]	0.60 [0.43,0.75]	0.71 [0.52,0.86]	0.76 [0.57,0.92]	0.79 [0.60,0.95]	0.81 [0.62,0.95]
	Δ	0.13 [0.05,0.23]	0.18 [0.06,0.31]	0.22 [0.07,0.38]	0.23 [0.07,0.40]	0.22 [0.05,0.40]	0.21 [0.05,0.39]	0.19 [0.05,0.38]
`Llama-3.1-8B` ($n=27$)	Cliff-del	0.54 [0.45,0.63]	0.73 [0.63,0.82]	0.87 [0.78,0.94]	0.95 [0.88,0.99]	0.98 [0.93,1.00]	1.00 [0.96,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.27 [0.18,0.37]	0.41 [0.28,0.53]	0.54 [0.39,0.67]	0.65 [0.50,0.78]	0.76 [0.59,0.87]	0.82 [0.66,0.93]	0.85 [0.70,0.96]
	Δ	0.27 [0.15,0.39]	0.33 [0.19,0.47]	0.34 [0.19,0.48]	0.29 [0.15,0.45]	0.22 [0.10,0.39]	0.18 [0.06,0.34]	0.15 [0.04,0.30]
`Llama-3.2-3B` ($n=21$)	Cliff-del	0.47 [0.35,0.61]	0.63 [0.51,0.75]	0.80 [0.70,0.88]	0.93 [0.86,0.97]	0.99 [0.96,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.15 [0.07,0.24]	0.24 [0.12,0.36]	0.35 [0.20,0.50]	0.45 [0.27,0.63]	0.53 [0.33,0.71]	0.59 [0.37,0.77]	0.67 [0.43,0.81]
	Δ	0.32 [0.19,0.47]	0.39 [0.24,0.55]	0.45 [0.28,0.62]	0.48 [0.30,0.66]	0.46 [0.27,0.66]	0.41 [0.23,0.63]	0.33 [0.19,0.62]
`Llama-3.2-1B` ($n=22$)	Cliff-del	0.43 [0.33,0.54]	0.62 [0.50,0.72]	0.79 [0.68,0.88]	0.91 [0.82,0.97]	0.97 [0.90,1.00]	0.99 [0.95,1.00]	1.00 [0.96,1.00]
	Cliff-keep	0.23 [0.14,0.33]	0.37 [0.24,0.50]	0.51 [0.35,0.66]	0.64 [0.46,0.79]	0.73 [0.54,0.88]	0.79 [0.59,0.93]	0.82 [0.64,0.96]
	Δ	0.20 [0.12,0.28]	0.25 [0.14,0.37]	0.28 [0.15,0.42]	0.27 [0.12,0.43]	0.24 [0.09,0.42]	0.21 [0.06,0.40]	0.18 [0.05,0.36]
`Gemma-3-4B` ($n=14$)	Cliff-del	0.48 [0.37,0.60]	0.69 [0.56,0.80]	0.86 [0.75,0.94]	0.96 [0.89,0.99]	0.99 [0.97,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.39 [0.27,0.53]	0.58 [0.43,0.72]	0.75 [0.62,0.87]	0.90 [0.79,0.96]	0.98 [0.92,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]
	Δ	0.09 [-0.01,0.19]	0.11 [-0.01,0.24]	0.10 [-0.01,0.23]	0.06 [-0.01,0.16]	0.02 [-0.01,0.07]	0.00 [0.00,0.02]	0.00 [0.00,0.00]

Table 10: **AIME 2025:** Cliff-del and Cliff-keep pass@k on incorrect traces. `n/e` indicates that the CI is not estimated since $n < 2$. – indicates that no traces were included in the analysis. Notations follow Table 8.

Model	Setup	$k = 1$	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 32$	$k = 64$
`Qwen3-8B` ($n = 3$)	Cliff-del	0.22 [0.14,0.32]	0.40 [0.25,0.54]	0.63 [0.43,0.79]	0.86 [0.67,0.96]	0.98 [0.87,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.12 [0.00,0.23]	0.21 [0.00,0.40]	0.35 [0.00,0.64]	0.52 [0.00,0.87]	0.64 [0.00,0.99]	0.67 [0.00,1.00]	0.67 [0.00,1.00]
	Δ	0.11 [-0.01,0.24]	0.19 [-0.02,0.42]	0.28 [-0.03,0.64]	0.34 [-0.05,0.87]	0.34 [-0.04,0.99]	0.33 [-0.01,1.00]	0.33 [0.00,1.00]
`Qwen3-4B` ($n = 4$)	Cliff-del	0.47 [0.24,0.74]	0.65 [0.41,0.89]	0.83 [0.63,0.97]	0.95 [0.83,1.00]	1.00 [0.95,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.35 [0.11,0.54]	0.54 [0.18,0.78]	0.69 [0.23,0.95]	0.75 [0.25,1.00]	0.75 [0.25,1.00]	0.75 [0.25,1.00]	0.75 [0.25,1.00]
	Δ	0.11 [-0.06,0.28]	0.12 [-0.06,0.29]	0.14 [-0.05,0.43]	0.21 [-0.02,0.61]	0.25 [0.00,0.73]	0.25 [0.00,0.75]	0.25 [0.00,0.75]
`Qwen3-0.6B` ($n = 1$)	Cliff-del	0.44 [n/e]	0.69 [n/e]	0.91 [n/e]	0.99 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.38 [n/e]	0.61 [n/e]	0.86 [n/e]	0.98 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.06 [n/e]	0.07 [n/e]	0.05 [n/e]	0.01 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]
`Llama-3.1-8B` ($n = 0$)	Cliff-del	–	–	–	–	–	–	–
	Cliff-keep	–	–	–	–	–	–	–
	Δ	–	–	–	–	–	–	–
`Llama-3.2-3B` ($n = 1$)	Cliff-del	0.25 [n/e]	0.44 [n/e]	0.69 [n/e]	0.92 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.23 [n/e]	0.42 [n/e]	0.67 [n/e]	0.90 [n/e]	0.99 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.02 [n/e]	0.02 [n/e]	0.03 [n/e]	0.02 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]
`Llama-3.2-1B` ($n = 0$)	Cliff-del	–	–	–	–	–	–	–
	Cliff-keep	–	–	–	–	–	–	–
	Δ	–	–	–	–	–	–	–
`Gemma-3-4B` ($n = 2$)	Cliff-del	0.37 [0.16,0.59]	0.58 [0.29,0.83]	0.78 [0.50,0.97]	0.93 [0.75,1.00]	0.99 [0.93,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.16 [0.10,0.23]	0.30 [0.19,0.42]	0.52 [0.35,0.66]	0.78 [0.59,0.89]	0.96 [0.83,0.99]	1.00 [0.97,1.00]	1.00 [1.00,1.00]
	Δ	0.20 [-0.03,0.45]	0.27 [-0.05,0.58]	0.26 [-0.07,0.56]	0.15 [-0.07,0.38]	0.03 [-0.03,0.16]	0.00 [0.00,0.03]	0.00 [0.00,0.00]

G.2 RESULTS FOR CORRECT TRACES

In addition to Section 4.2, we report the complementary analysis on correct traces. Tables 11 to 13 show the corresponding results for GSM1K, MATH500, and AIME 2025, respectively. Cliff-del yields equal or higher mean pass@k compared to Cliff-keep in most settings, with only a few exceptions. Overall, these results show that cliff tokens reduce success probability even in traces that reach the correct answer.

Table 11: **GSM1K:** Cliff-del and Cliff-keep pass@k on correct traces. Notations follow Table 8.

Model	Setup	$k=1$	$k=2$	$k=4$	$k=8$	$k=16$	$k=32$	$k=64$
`Qwen3-8B` ($n=2$)	Cliff-del	0.72 [0.54,0.88]	0.91 [0.79,0.99]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.73 [0.56,0.88]	0.91 [0.80,0.99]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	-0.01 [-0.12,0.09]	-0.01 [-0.08,0.07]	0.00 [-0.03,0.02]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Qwen3-4B` ($n=2$)	Cliff-del	0.72 [0.48,0.94]	0.89 [0.73,1.00]	0.98 [0.92,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.50 [0.06,0.94]	0.59 [0.12,1.00]	0.67 [0.23,1.00]	0.78 [0.40,1.00]	0.92 [0.63,1.00]	0.99 [0.87,1.00]	1.00 [1.00,1.00]
	Δ	0.22 [-0.05,0.50]	0.30 [-0.01,0.68]	0.31 [0.00,0.73]	0.22 [0.00,0.60]	0.08 [0.00,0.38]	0.01 [0.00,0.13]	0.00 [0.00,0.00]
`Qwen3-0.6B` ($n=5$)	Cliff-del	0.67 [0.47,0.85]	0.85 [0.69,0.98]	0.96 [0.88,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.63 [0.41,0.83]	0.81 [0.62,0.97]	0.93 [0.82,1.00]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.04 [-0.03,0.11]	0.04 [-0.02,0.12]	0.03 [-0.02,0.10]	0.01 [-0.01,0.04]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Llama-3.1-8B` ($n=10$)	Cliff-del	0.71 [0.57,0.85]	0.87 [0.75,0.96]	0.96 [0.88,1.00]	0.99 [0.96,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.51 [0.38,0.64]	0.73 [0.58,0.85]	0.89 [0.78,0.96]	0.97 [0.92,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.20 [0.09,0.31]	0.15 [0.05,0.26]	0.07 [0.00,0.17]	0.02 [-0.01,0.07]	0.00 [0.00,0.01]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Llama-3.2-3B` ($n=6$)	Cliff-del	0.54 [0.38,0.73]	0.75 [0.59,0.89]	0.91 [0.81,0.98]	0.99 [0.95,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.48 [0.32,0.64]	0.69 [0.52,0.84]	0.87 [0.74,0.96]	0.97 [0.90,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.07 [-0.13,0.27]	0.06 [-0.14,0.25]	0.04 [-0.11,0.19]	0.02 [-0.04,0.09]	0.00 [0.00,0.02]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Llama-3.2-1B` ($n=7$)	Cliff-del	0.67 [0.57,0.77]	0.88 [0.80,0.94]	0.98 [0.95,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.42 [0.27,0.55]	0.63 [0.42,0.79]	0.81 [0.55,0.95]	0.88 [0.64,1.00]	0.92 [0.71,1.00]	0.97 [0.79,1.00]	1.00 [0.86,1.00]
	Δ	0.25 [0.12,0.38]	0.25 [0.09,0.44]	0.17 [0.03,0.42]	0.12 [0.00,0.36]	0.08 [0.00,0.29]	0.04 [0.00,0.21]	0.00 [0.00,0.14]
`Gemma-3-4B` ($n=7$)	Cliff-del	0.70 [0.60,0.80]	0.90 [0.83,0.95]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.57 [0.45,0.71]	0.79 [0.68,0.89]	0.94 [0.89,0.98]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.13 [0.03,0.24]	0.11 [0.03,0.20]	0.04 [0.01,0.09]	0.00 [0.00,0.02]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]

Table 12: **MATH500:** Cliff-del and Cliff-keep pass@k on correct traces. Notations follow Table 8.

Model	Setup	$k=1$	$k=2$	$k=4$	$k=8$	$k=16$	$k=32$	$k=64$
`Qwen3-8B` ($n=13$)	Cliff-del	0.68 [0.58,0.77]	0.87 [0.79,0.93]	0.97 [0.92,0.99]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.63 [0.54,0.71]	0.85 [0.77,0.90]	0.97 [0.92,0.99]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.05 [-0.03,0.14]	0.03 [-0.05,0.11]	0.01 [-0.04,0.05]	0.00 [-0.02,0.01]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Qwen3-4B` ($n=9$)	Cliff-del	0.62 [0.48,0.75]	0.82 [0.69,0.92]	0.94 [0.87,0.99]	0.99 [0.97,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.61 [0.48,0.74]	0.81 [0.69,0.92]	0.94 [0.86,0.99]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.01 [-0.08,0.12]	0.00 [-0.06,0.07]	0.00 [-0.05,0.05]	0.00 [-0.02,0.02]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Qwen3-0.6B` ($n=10$)	Cliff-del	0.50 [0.38,0.63]	0.72 [0.59,0.83]	0.89 [0.80,0.96]	0.98 [0.93,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.45 [0.32,0.59]	0.65 [0.50,0.79]	0.84 [0.71,0.93]	0.95 [0.86,0.99]	0.99 [0.96,1.00]	1.00 [0.99,1.00]	1.00 [1.00,1.00]
	Δ	0.05 [-0.04,0.15]	0.06 [-0.03,0.17]	0.06 [-0.01,0.14]	0.03 [-0.01,0.09]	0.01 [0.00,0.04]	0.00 [0.00,0.01]	0.00 [0.00,0.00]
`Llama-3.1-8B` ($n=15$)	Cliff-del	0.64 [0.52,0.76]	0.82 [0.73,0.91]	0.94 [0.89,0.98]	0.99 [0.97,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.48 [0.34,0.62]	0.66 [0.50,0.80]	0.81 [0.65,0.93]	0.89 [0.76,0.98]	0.95 [0.84,1.00]	0.98 [0.90,1.00]	1.00 [0.93,1.00]
	Δ	0.16 [0.03,0.30]	0.16 [0.03,0.31]	0.14 [0.02,0.28]	0.10 [0.01,0.22]	0.05 [0.00,0.16]	0.02 [0.00,0.10]	0.00 [0.00,0.07]
`Llama-3.2-3B` ($n=8$)	Cliff-del	0.61 [0.43,0.75]	0.80 [0.60,0.93]	0.90 [0.72,0.99]	0.95 [0.81,1.00]	0.98 [0.90,1.00]	1.00 [0.97,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.50 [0.32,0.66]	0.70 [0.46,0.87]	0.82 [0.57,0.98]	0.87 [0.62,1.00]	0.88 [0.63,1.00]	0.88 [0.63,1.00]	0.88 [0.63,1.00]
	Δ	0.10 [0.00,0.21]	0.10 [0.01,0.20]	0.08 [0.01,0.18]	0.08 [0.00,0.23]	0.11 [0.00,0.32]	0.12 [0.00,0.37]	0.13 [0.00,0.38]
`Llama-3.2-1B` ($n=9$)	Cliff-del	0.73 [0.59,0.86]	0.89 [0.78,0.96]	0.97 [0.91,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.56 [0.44,0.67]	0.78 [0.66,0.88]	0.93 [0.85,0.98]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.17 [0.06,0.30]	0.11 [0.04,0.20]	0.04 [0.01,0.09]	0.01 [-0.01,0.03]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Gemma-3-4B` ($n=7$)	Cliff-del	0.80 [0.70,0.87]	0.95 [0.90,0.98]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.66 [0.54,0.78]	0.86 [0.77,0.94]	0.97 [0.94,0.99]	1.00 [0.99,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	0.14 [0.03,0.25]	0.09 [0.02,0.17]	0.02 [0.00,0.06]	0.00 [0.00,0.01]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]

Table 13: **AIME 2025:** Cliff-del and Cliff-keep pass@k on correct traces. `n/e` indicates that the CI is not estimated since $n < 2$. – indicates that no traces were included in the analysis. Notations follow Table 8.

Model	Setup	$k = 1$	$k = 2$	$k = 4$	$k = 8$	$k = 16$	$k = 32$	$k = 64$
`Qwen3-8B` ($n = 2$)	Cliff-del	0.61 [0.32,0.88]	0.80 [0.54,0.99]	0.93 [0.79,1.00]	0.99 [0.96,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Cliff-keep	0.62 [0.38,0.84]	0.82 [0.62,0.98]	0.95 [0.86,1.00]	1.00 [0.98,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]	1.00 [1.00,1.00]
	Δ	-0.01 [-0.15,0.12]	-0.03 [-0.17,0.07]	-0.03 [-0.13,0.04]	-0.01 [-0.04,0.01]	0.00 [0.00,0.00]	0.00 [0.00,0.00]	0.00 [0.00,0.00]
`Qwen3-4B` ($n = 1$)	Cliff-del	0.63 [n/e]	0.86 [n/e]	0.98 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.25 [n/e]	0.44 [n/e]	0.69 [n/e]	0.92 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.38 [n/e]	0.42 [n/e]	0.29 [n/e]	0.09 [n/e]	0.01 [n/e]	0.00 [n/e]	0.00 [n/e]
`Qwen3-0.6B` ($n = 1$)	Cliff-del	0.48 [n/e]	0.74 [n/e]	0.94 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.44 [n/e]	0.69 [n/e]	0.91 [n/e]	0.99 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.05 [n/e]	0.05 [n/e]	0.03 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]
`Llama-3.1-8B` ($n = 1$)	Cliff-del	0.48 [n/e]	0.74 [n/e]	0.94 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.38 [n/e]	0.61 [n/e]	0.86 [n/e]	0.98 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.11 [n/e]	0.13 [n/e]	0.08 [n/e]	0.01 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]
`Llama-3.2-3B` ($n = 0$)	Cliff-del	–	–	–	–	–	–	–
	Cliff-keep	–	–	–	–	–	–	–
	Δ	–	–	–	–	–	–	–
`Llama-3.2-1B` ($n = 0$)	Cliff-del	–	–	–	–	–	–	–
	Cliff-keep	–	–	–	–	–	–	–
	Δ	–	–	–	–	–	–	–
`Gemma-3-4B` ($n = 1$)	Cliff-del	0.97 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Cliff-keep	0.84 [n/e]	0.98 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]	1.00 [n/e]
	Δ	0.13 [n/e]	0.02 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]	0.00 [n/e]

H TOKEN ENTROPY DISTRIBUTION

As illustrated in Figure 6, the overall token entropy during mathematical reasoning exhibits a high probability density near zero, peaking at $H \approx 0$. In contrast, the distribution for cliff tokens shows a lower density in this low-entropy regime. Furthermore, the cliff token entropy displays a heavy-tailed distribution. This empirical evidence indicates that models frequently sample cliff tokens under elevated uncertainty.

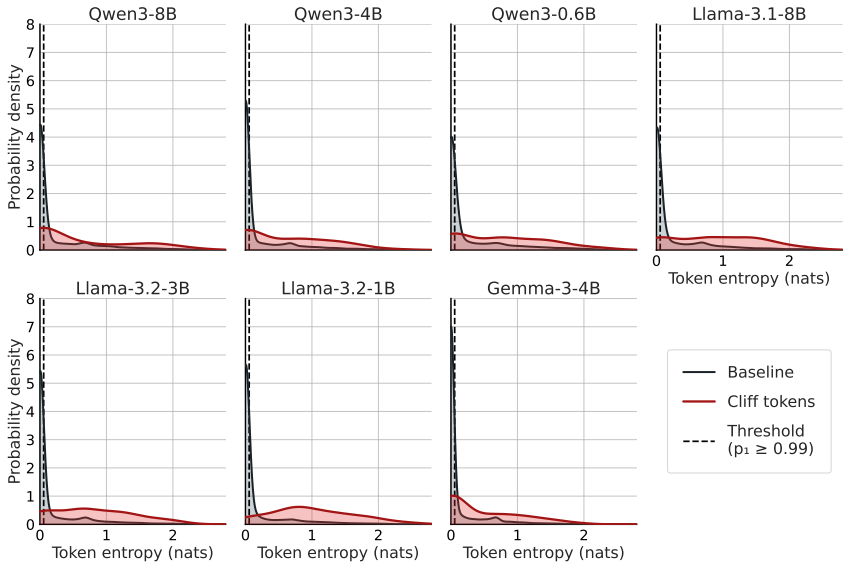


Figure 6: Probability density distributions of token entropy across the seven models. Token entropy is aggregated across the GSM1K, MATH500, and AIME 2025 datasets. The baseline represents the token entropy computed over all generated tokens within the reasoning traces, whereas the cliff token distribution represents the token entropy at the cliff token positions.

I GREEDY TOKEN RATIOS

As detailed in Table 14, the greedy-token ratio for cliff token is significantly lower than the overall baseline across all evaluated models. This reduction is largest in the Llama variants, whose cliff-token greedy ratios fall below 0.5. In contrast, a majority of cliff tokens remain greedy for `Qwen3` family and `Gemma-3-4B`.

Table 14: Comparison of greedy token sampling rates. Ratios are aggregated across the GSM1K, MATH500, and AIME 2025 datasets. The baseline represents the average greedy token ratio computed over all generated tokens within the reasoning traces.

Model	Cliff Tokens	Baseline
`Qwen3-8B`	0.8229	0.9685
`Qwen3-4B`	0.8056	0.9747
`Qwen3-0.6B`	0.7177	0.9650
`Llama-3.1-8B`	0.4724	0.9516
`Llama-3.2-3B`	0.4211	0.9599
`Llama-3.2-1B`	0.3784	0.9636
`Gemma-3-4B`	0.7500	0.9732

J ROBUSTNESS OF THE CLIFF TAXONOMY TO THE ENTROPY THRESHOLD

To evaluate the robustness of the cliff taxonomy, we conduct an ablation study using alternative token entropy thresholds calculated on binary entropy with $p_1 \in \{0.90, 0.95, 0.999\}$ in Section 5.1. As detailed in Table 15, the qualitative trends reported in Section 5.2 remain consistent across all thresholds. The deterministic cliff proportion is lower than the all-token proportion, while sampled-off cliffs occur at a higher proportion across all models. Furthermore, the scaling behavior within the `Qwen3` and `Llama-3` families is preserved across all settings, except for sampled-off cliffs on `Llama-3.2-3B`. These results indicate that the findings on cliff tokens are not sensitive to the specific boundary of the entropy split.

Table 15: Distribution of cliff types under alternative entropy splits ($p_1 \in \{0.90, 0.95, 0.999\}$). Column definitions follow Table 1.

	Cliff proportion (%)			**All-token proportion (%)**		
Model	D	U	S	D	U	S
$H_b \approx 0.325$ **nats** ($p_1 = 0.90$)						
`Qwen3-8B`	56.2	26.0	17.7	78.4	18.5	3.1
`Qwen3-4B`	47.2	33.3	19.4	81.9	15.5	2.5
`Qwen3-0.6B`	37.9	33.9	28.2	75.5	21.0	3.5
`Llama-3.1-8B`	25.2	22.0	50.4	77.8	17.3	4.8
`Llama-3.2-3B`	21.1	21.1	51.6	82.3	13.7	3.9
`Llama-3.2-1B`	13.5	24.3	60.4	84.3	12.1	3.6
`Gemma-3-4B`	51.2	23.8	17.9	86.2	11.2	2.6
$H_b \approx 0.199$ **nats** ($p_1 = 0.95$)						
`Qwen3-8B`	53.1	29.2	17.7	75.2	21.6	3.1
`Qwen3-4B`	44.4	36.1	19.4	79.2	18.3	2.5
`Qwen3-0.6B`	34.7	37.1	28.2	71.9	24.6	3.5
`Llama-3.1-8B`	24.4	22.8	52.0	74.5	20.7	4.8
`Llama-3.2-3B`	18.9	23.2	51.6	79.6	16.4	4.0
`Llama-3.2-1B`	10.8	27.0	60.4	81.6	14.7	3.6
`Gemma-3-4B`	51.2	23.8	19.0	83.4	13.9	2.7
$H_b \approx 0.008$ **nats** ($p_1 = 0.999$)						
`Qwen3-8B`	46.9	35.4	17.7	61.9	35.0	3.1
`Qwen3-4B`	38.9	41.7	19.4	67.4	30.1	2.5
`Qwen3-0.6B`	20.2	51.6	28.2	52.4	44.1	3.5
`Llama-3.1-8B`	18.1	29.1	52.8	52.0	43.1	4.8
`Llama-3.2-3B`	15.8	26.3	56.8	61.3	34.6	4.0
`Llama-3.2-1B`	8.1	29.7	62.2	56.7	39.7	3.6
`Gemma-3-4B`	42.9	32.1	22.6	70.4	26.9	2.7

K COUNTERFACTUAL ANALYSIS ON SAMPLED-OFF CLIFFS

To verify that sampled-off cliffs are induced by stochastic sampling noise rather than an inherent systemic preference toward failure, we conduct a counterfactual analysis on `Qwen3-8B`. Specifically, we override the sampled-off cliff token with its corresponding greedy token and measure the resulting success probability.

As illustrated in Figure 7, the vast majority of data points lie significantly above the $y = x$ reference line. Out of all evaluated instances, we observe only one case where the success probability remains unchanged and one case where it slightly decreases. In all other instances, substituting with the greedy token successfully increases the success probability. This confirms that sampled-off cliffs are indeed suboptimal choices drawn by chance, and replacing them with the greedy token effectively mitigates the success probability drop.

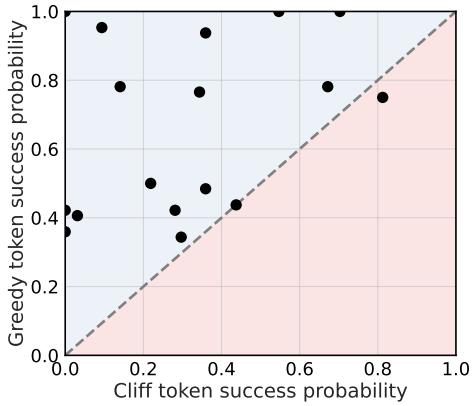


Figure 7: Counterfactual analysis on sampled-off cliffs using `Qwen3-8B`. The experiment evaluates 17 sampled-off cliffs identified across the GSM1K, MATH500, and AIME 2025 datasets. Greedy-token success probability refers to the success probability when these sampled-off cliffs are replaced with greedy tokens. Points above the diagonal dashed line indicate an increase in success probability.

L CROSS-SCALE CLIFF PROBABILITY MASS TRANSFER

While Section 5.2 shows macro-level distributional shifts across model families and scales, it remains unclear whether scale-dependent cliff behavior persists at exact token positions within a family. To investigate this token-level transferability, we conduct cross-scale transfer experiments. Given a cliff token c_{t^*} identified by a source model, we evaluate a target model on the same source-generated cliff-del prefix ($\boldsymbol{x} \oplus \boldsymbol{c}_{<t^*}$). We then observe how the cliff probability mass at this decoding step t^* shifts between the two models. Because exact position alignment requires a shared tokenizer, we restrict this evaluation to two models of different sizes within the same family: `Qwen3-0.6B` and `Qwen3-8B`. Our experiments show distinct distributions of cliff probability mass difference (Δ) for each type (Figure 8):

- *Deterministic cliffs: Scale-invariance*. We observe that the difference in cliff probability mass is near zero ($\Delta \approx 0$) in both `Qwen3-0.6B` $\rightarrow$ `8B` and `8B` $\rightarrow$ `0.6B` transfers. At these exact reasoning positions, both models sample the identical deterministic cliffs. Specifically, at 44 out of 46 deterministic cliff positions identified by `Qwen3-8B`, `Qwen3-0.6B` also samples the same cliff tokens. Conversely, at all 37 deterministic cliff positions identified by `Qwen3-0.6B`, `Qwen3-8B` mirrors this selection (detailed in Appendix M). The bidirectional overlap indicates that deterministic cliffs are largely scale-invariant within the `Qwen3` family. Given the same reasoning prefix, both model sizes select the same failure-triggering token, suggesting that these cliff tokens reflect a shared family-level bias.
- *Uncertain cliffs: Model-specific failure patterns*. Regardless of the transfer direction, the probability mass of these cliff tokens decreases during cross-model transfers ($\Delta \approx -0.13$ and -0.10). This overall drop suggests that uncertain cliffs reflect model-specific failure patterns, rather than shared reasoning bottlenecks across the `Qwen3` family. Consequently, uncertain cliffs expose the unique failure spots of individual models.
- *Sampled-off cliffs: Scale-asymmetry*. We observe a clear asymmetry in cliff probability mass shifts: the mean Δ is negative for `Qwen3-0.6B` $\rightarrow$ `8B`, but positive for `8B` $\rightarrow$ `0.6B`. Positions identified as sampled-off cliffs by `Qwen3-8B` act as high-probability cliff regions for `Qwen3-0.6B`. This connects to the scaling trend in Section 5.2, where the sampled-off cliff proportion decreases (28.2 $\rightarrow$ 17.7) with `Qwen3` model scaling.

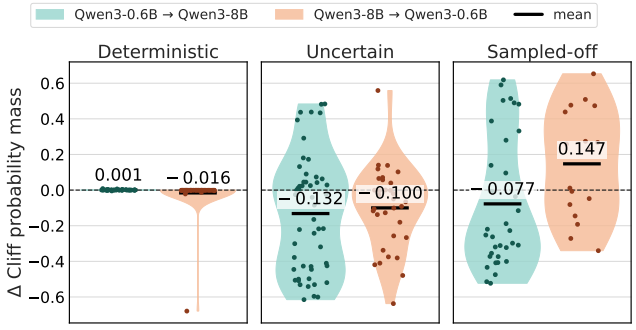


Figure 8: Cliff probability mass shifts (Δ) at identified cliff positions upon cross-scale transfer between `Qwen3-0.6B` and `Qwen3-8B`. Deterministic cliffs are scale-invariant ($\Delta \approx 0$). Uncertain cliffs show overall mass decrease. Sampled-off cliffs exhibit scale-asymmetry.

M RANK DIFFERENCE OF CLIFF TOKEN IN CROSS-MODEL EXPERIMENT

This appendix complements Appendix L with a token-position-level view of cross-model transfer. For each cliff token c_{t^*} identified by the source model, we feed the cliff-del prefix $\boldsymbol{x} \oplus \boldsymbol{c}_{<t^*}$ to the target model and record the rank of c_{t^*} within the target's top-20 candidates at step t^*. Tokens outside the top-20 are placed at rank 21. Figures 9 to 11 show the resulting rank shifts for deterministic, uncertain, and sampled-off cliffs, respectively.

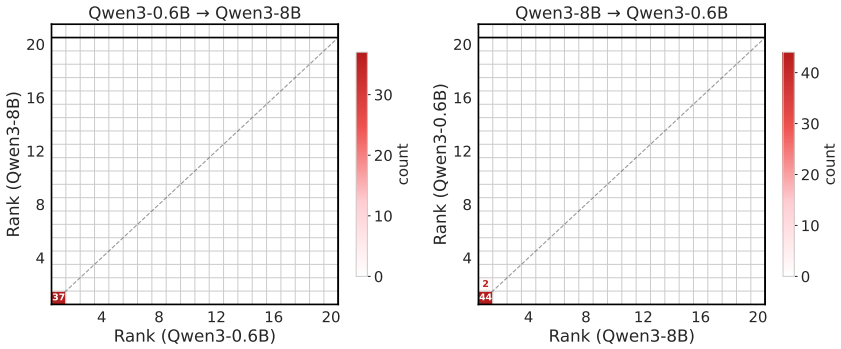


Figure 9: Rank transfer of *deterministic cliffs* between `Qwen3-0.6B` and `Qwen3-8B`. The cliff token's rank is preserved at 1 in both transfer directions, with only 2 deterministic cliffs moving to rank 2 in the `Qwen3-8B` → `Qwen3-0.6B` direction.

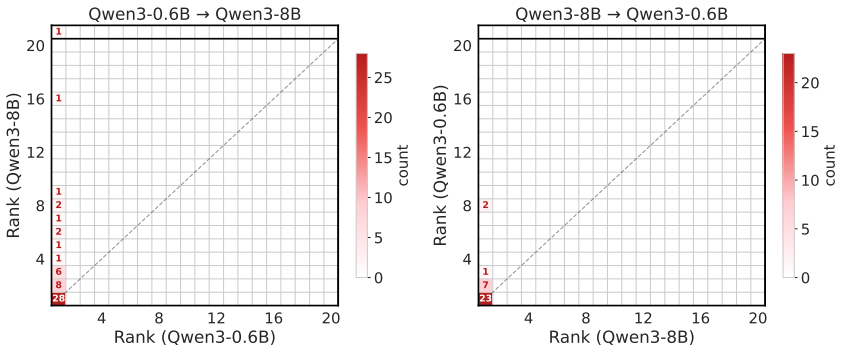


Figure 10: Rank transfer of *uncertain cliffs*. In the `Qwen3-0.6B` → `Qwen3-8B` direction, $24/52$ cliff tokens shifted away from rank 1, while in the `Qwen3-8B` → `Qwen3-0.6B` direction, $10/33$ cliff tokens shifted away from rank 1.

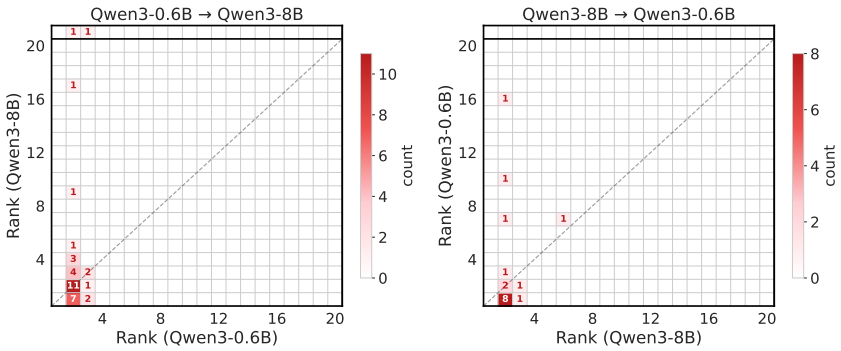


Figure 11: Rank transfer of *sampled-off cliffs*. The rank shifts asymmetrically across transfer directions. In the `Qwen3-0.6B` → `Qwen3-8B` direction, the rank is preserved, increased, or decreased in roughly comparable proportions, whereas in the `Qwen3-8B` → `Qwen3-0.6B` direction, $10/17$ cliff tokens move to a lower rank index (higher probability) in the target model.

N FULL RESULTS OF CLIFF-DPO

We trained five variants of Cliff-DPO for each model: *deterministic*, *uncertain*, *sampled-off*, *uncertain + sampled-off*, and *all*. Each name indicates the subset of cliff tokens used during training. As shown in Table 16, for all models, the *uncertain* and *sampled-off* variants outperform the *deterministic* variant across GSM8K, GSM1K, and MATH500, as reported in Section 6.2. Furthermore, the *all* variant, which adds the deterministic subset to the *uncertain + sampled-off* variant, does not consistently outperform the *uncertain + sampled-off* variant. These results suggest that incorporating the deterministic subset provides little additional benefit for post-training.

Table 16: Cliff-DPO variants and baselines on three models. The table reports mean accuracy or avg@64, along with the sample standard error. *Updated tokens* counts gradient-updated token positions. **Bold** and underline mark the best and second-best results within each model on GSM8K, GSM1K, and MATH500. AIME 2025 is not ranked.

	Mean accuracy			avg@64	
Method	GSM8K	GSM1K	MATH500	AIME 2025	**Updated tokens**
`Llama-3.1-8B`					
Base	$82.2_{\pm 0.20}$	$78.2_{\pm 0.07}$	$48.1_{\pm 0.29}$	$0.7_{\pm 0.20}$	–
+ DPO	$83.2_{\pm 0.25}$	$78.4_{\pm 0.19}$	$47.6_{\pm 0.12}$	$0.6_{\pm 0.03}$	1,952,191
+ cDPO	$\mathbf{84.7}_{\pm 0.35}$	$\mathbf{81.7}_{\pm 0.34}$	$49.1_{\pm 1.39}$	$0.8_{\pm 0.21}$	3,232,251
+ Cliff-DPO					
deterministic	$82.1_{\pm 0.31}$	$78.1_{\pm 0.30}$	$48.5_{\pm 0.88}$	$0.6_{\pm 0.09}$	5,618
uncertain	$\underline{84.6}_{\pm 0.15}$	$79.6_{\pm 0.07}$	$\mathbf{50.8}_{\pm 0.72}$	$0.5_{\pm 0.03}$	20,848
sampled-off	$83.1_{\pm 0.14}$	$78.3_{\pm 0.15}$	$49.1_{\pm 1.09}$	$0.5_{\pm 0.14}$	13,236
uncertain + sampled-off	$83.9_{\pm 0.24}$	$\underline{79.8}_{\pm 0.31}$	$49.8_{\pm 0.42}$	$0.6_{\pm 0.08}$	34,084
all	$84.5_{\pm 0.25}$	$\underline{79.8}_{\pm 0.13}$	$\underline{50.5}_{\pm 0.07}$	$0.5_{\pm 0.02}$	39,708
`Llama-3.2-1B`					
Base	$39.3_{\pm 0.17}$	$36.0_{\pm 0.54}$	$25.4_{\pm 0.42}$	$0.2_{\pm 0.08}$	–
+ DPO	$40.8_{\pm 0.33}$	$35.7_{\pm 0.12}$	$27.5_{\pm 0.44}$	$0.2_{\pm 0.05}$	2,264,627
+ cDPO	$45.1_{\pm 0.56}$	$\mathbf{41.4}_{\pm 0.43}$	$26.3_{\pm 0.18}$	$0.2_{\pm 0.05}$	7,264,379
+ Cliff-DPO					
deterministic	$41.5_{\pm 0.50}$	$36.2_{\pm 0.59}$	$26.6_{\pm 0.40}$	$0.0_{\pm 0.02}$	3,900
uncertain	$44.5_{\pm 0.28}$	$37.4_{\pm 0.17}$	$27.9_{\pm 0.33}$	$0.1_{\pm 0.05}$	14,886
sampled-off	$\mathbf{45.6}_{\pm 0.46}$	$\underline{40.4}_{\pm 0.70}$	$\mathbf{29.6}_{\pm 0.46}$	$0.1_{\pm 0.05}$	47,266
uncertain + sampled-off	$\underline{45.3}_{\pm 0.48}$	$39.4_{\pm 0.41}$	$\underline{29.2}_{\pm 0.76}$	$0.0_{\pm 0.03}$	62,152
all	$\mathbf{45.6}_{\pm 0.37}$	$39.0_{\pm 0.14}$	$28.1_{\pm 0.24}$	$0.2_{\pm 0.03}$	66,388
`Qwen3-0.6B`					
Base	$61.5_{\pm 0.27}$	$57.0_{\pm 0.06}$	$51.6_{\pm 0.00}$	$2.0_{\pm 1.04}$	–
+ DPO	$63.0_{\pm 0.60}$	$56.6_{\pm 0.45}$	$52.6_{\pm 0.80}$	$2.4_{\pm 0.10}$	2,862,845
+ cDPO	$\underline{66.4}_{\pm 0.48}$	$61.4_{\pm 0.53}$	$54.0_{\pm 0.61}$	$1.6_{\pm 1.61}$	5,829,052
+ Cliff-DPO					
deterministic	$63.7_{\pm 0.18}$	$57.3_{\pm 0.48}$	$51.9_{\pm 1.51}$	$2.7_{\pm 0.09}$	5,538
uncertain	$\mathbf{66.8}_{\pm 0.07}$	$62.0_{\pm 0.17}$	$53.7_{\pm 0.47}$	$2.7_{\pm 1.35}$	18,122
sampled-off	$\mathbf{66.8}_{\pm 0.24}$	$\underline{62.9}_{\pm 0.21}$	$\underline{54.9}_{\pm 0.93}$	$3.7_{\pm 0.19}$	14,794
uncertain + sampled-off	$66.3_{\pm 0.47}$	$\mathbf{63.0}_{\pm 0.28}$	$\mathbf{56.1}_{\pm 0.59}$	$4.7_{\pm 0.27}$	32,916
all	$65.6_{\pm 0.07}$	$61.0_{\pm 0.23}$	$54.5_{\pm 0.85}$	$1.3_{\pm 1.34}$	38,454

O CLIFF-TOKEN PROBABILITY BEFORE AND AFTER CLIFF-DPO

To analyze the cliff-probability shift after Cliff-DPO discussed in Section 6.2, we apply the cliff token identification procedure of Section 3.3 to pre-identified cliff token positions in GSM1K, MATH500, and AIME 2025 traces. For each cliff token at position t, we measure its generation probability to evaluate whether the model reduces the probability of selecting it after Cliff-DPO. This procedure is performed on 127, 111, and 124 held-out cliff positions for `Llama-3.1-8B`, `Llama-3.2-1B`, and `Qwen3-0.6B`, respectively. We average the results over the three training seeds for each Cliff-DPO variant. Table 17 reports the cliff-token probability shifts for all variants across the three cliff types.

For deterministic cliff tokens, the initial probability is 99.9%, and no Cliff-DPO variant changes it by more than 0.24 pp. For uncertain cliff tokens, all Cliff-DPO variants reduce the cliff-token probability. For sampled-off cliff tokens, only the *sampled-off* variant consistently reduces the cliff-token probability.

Table 17: Average cliff-token probability before and after Cliff-DPO. Rows denote training variants and columns denote evaluation cliff types. Changes from the base model are in percentage points (pp). Darker gray cells mark the diagonal entries reported in Fig. 5.

	Average cliff-token probability (%)		
Method	Deterministic	Uncertain	Sampled-off
`Llama-3.1-8B`			
Base	99.9	56.4	20.1
+ Cliff-DPO			
deterministic	$99.9_{(-0.00)}$	$56.2_{(-0.25)}$	$20.4_{(+0.26)}$
uncertain	$99.9_{(-0.05)}$	$54.9_{(-1.56)}$	$20.4_{(+0.28)}$
sampled-off	$99.9_{(-0.01)}$	$55.7_{(-0.73)}$	$20.0_{(-0.09)}$
`Llama-3.2-1B`			
Base	99.9	62.2	21.7
+ Cliff-DPO			
deterministic	$99.9_{(-0.01)}$	$61.9_{(-0.35)}$	$21.8_{(+0.08)}$
uncertain	$99.9_{(-0.05)}$	$59.6_{(-2.67)}$	$21.6_{(-0.08)}$
sampled-off	$99.9_{(-0.01)}$	$55.6_{(-6.61)}$	$19.0_{(-2.72)}$
`Qwen3-0.6B`			
Base	99.9	63.0	28.6
+ Cliff-DPO			
deterministic	$99.8_{(-0.06)}$	$62.2_{(-0.80)}$	$28.8_{(+0.16)}$
uncertain	$99.6_{(-0.24)}$	$58.0_{(-4.99)}$	$26.9_{(-1.74)}$
sampled-off	$99.7_{(-0.14)}$	$59.1_{(-3.86)}$	$26.6_{(-2.02)}$

P CLIFF-DPO ABLATION UNDER MATCHED UPDATE-TOKEN BUDGET

To isolate the effect of cliff type from that of training-data quantity, we subsample each single-type training set to match the number of gradient-updated tokens in the *deterministic* subset: 5,618 for `Llama-3.1-8B`, 3,900 for `Llama-3.2-1B`, and 5,538 for `Qwen3-0.6B`. The *deterministic* subset is fully used, whereas the *uncertain* and *sampled-off* subsets are subsampled using a random seed of 42. Table 18 reports the results.

Under this matched-token setting, the main finding of Section 6.2 continues to hold for `Llama-3.1-8B` and `Qwen3-0.6B`: both the *uncertain* and *sampled-off* variants outperform the *deterministic* variant on GSM8K, GSM1K, and MATH500. `Llama-3.2-1B` is an exception: the *deterministic* variant performs best on GSM8K and GSM1K. This may reflect its small number of training tokens (3,900 tokens). At their full budgets in Section 6.2, the *uncertain* and *sampled-off* variants gain a further 3.8 and 5.4 points on GSM8K, respectively.

Table 18: Token-matched ablation of Cliff-DPO variants on three models. Each variant is trained on a single cliff type, with the update-token budget subsampled within each model to match the *deterministic* subset. Notation follows Table 16.

	Mean accuracy			avg@64	
Method	GSM8K	GSM1K	MATH500	AIME 2025	**Updated tokens**
`Llama-3.1-8B`					
Base	$\underline{82.2}_{\pm 0.20}$	$\underline{78.2}_{\pm 0.07}$	$48.1_{\pm 0.29}$	$0.7_{\pm 0.20}$	–
+ Cliff-DPO					
deterministic	$82.1_{\pm 0.31}$	$78.1_{\pm 0.30}$	$48.5_{\pm 0.88}$	$0.6_{\pm 0.09}$	5,618
uncertain	$\mathbf{83.0}_{\pm 0.24}$	$\mathbf{78.5}_{\pm 0.10}$	$\underline{48.7}_{\pm 0.37}$	$0.5_{\pm 0.31}$	5,618
sampled-off	$\mathbf{83.0}_{\pm 0.14}$	$\underline{78.2}_{\pm 0.18}$	$\mathbf{48.9}_{\pm 0.74}$	$0.5_{\pm 0.27}$	5,618
`Llama-3.2-1B`					
Base	$39.3_{\pm 0.17}$	$\underline{36.0}_{\pm 0.54}$	$25.4_{\pm 0.42}$	$0.2_{\pm 0.08}$	–
+ Cliff-DPO					
deterministic	$\mathbf{41.5}_{\pm 0.50}$	$\mathbf{36.2}_{\pm 0.59}$	$26.6_{\pm 0.40}$	$0.0_{\pm 0.02}$	3,900
uncertain	$\underline{40.7}_{\pm 0.24}$	$34.7_{\pm 0.05}$	$\mathbf{28.4}_{\pm 0.70}$	$0.1_{\pm 0.03}$	3,900
sampled-off	$40.2_{\pm 0.21}$	$35.6_{\pm 0.20}$	$\underline{27.6}_{\pm 0.92}$	$0.1_{\pm 0.03}$	3,900
`Qwen3-0.6B`					
Base	$61.5_{\pm 0.27}$	$57.0_{\pm 0.06}$	$51.6_{\pm 0.00}$	$2.0_{\pm 1.04}$	–
+ Cliff-DPO					
deterministic	$63.7_{\pm 0.18}$	$57.3_{\pm 0.48}$	$\underline{51.9}_{\pm 1.51}$	$\underline{2.7}_{\pm 0.09}$	5,538
uncertain	$\mathbf{64.5}_{\pm 0.15}$	$\underline{58.3}_{\pm 0.47}$	$\mathbf{53.1}_{\pm 0.24}$	$\underline{2.7}_{\pm 1.41}$	5,538
sampled-off	$\underline{64.0}_{\pm 0.05}$	$\mathbf{58.6}_{\pm 0.29}$	$\mathbf{53.1}_{\pm 0.18}$	$\mathbf{3.1}_{\pm 1.83}$	5,538

Q TRAINING AND EVALUATION CONFIGURATION

Experiments in Section 6 use `Llama-3.1-8B` (Grattafiori et al., 2024), `Llama-3.2-1B` (Grattafiori et al., 2024), and `Qwen3-0.6B` (Yang et al., 2025) as base models, and all preference data is constructed from the GSM8K (Cobbe et al., 2021) training set (7,473 problems). All training is conducted in BF16 with seeds `42`, `43`, and `44`.

Q.1 DPO

The DPO baseline follows Rafailov et al. (2023). We sample 64 traces per problem from the base model on the GSM8K training set and pair, for each problem, one correct trace (chosen) with one incorrect trace (rejected). This produces 4,006 trace-level preference pairs for `Llama-3.1-8B`, 6,797 for `Llama-3.2-1B`, and 5,566 for `Qwen3-0.6B`. The remaining problems whose 64 traces are all correct or all incorrect are excluded from the dataset.

Training hyperparameters. Sigmoid DPO loss with $\beta = 0.1$ and no label smoothing. Training runs for 1 epoch with learning rate 1e$-$6, cosine decay, and warmup ratio 0.1. The batch size is 64. Gradient clipping uses a max norm of 1.0; weight decay is disabled. LoRA: rank $r = 32$, $\alpha = 64$, dropout 0.05, applied to all seven attention and feed-forward projection modules: `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`. We additionally verified that aligning DPO's training configuration with that of Cliff-DPO degrades its performance; we therefore report DPO under configuration recommended by Rafailov et al. (2023).

Q.2 cDPO

We reproduce the five-step pipeline of Lin et al. (2025) without modification, except that the models are replaced with the three base models used in this work, for consistency with our other baselines. The contrastive logit formulation, $\beta = 1.0$ in the Contrastive Estimation (CE) score s_t, the `only_neg=True` setting, and the per-token $(1 - s_t)$ weighting on the rejected side all follow the exact configurations detailed in Lin et al. (2025).

Auxiliary contrastive SFT models. The positive SFT adapter π_+ is trained on one correct trace per problem. The negative SFT adapter π_- is trained on all incorrect traces from the problems that have both correct and incorrect rollouts, since multiple distinct failure modes exist for the same problem. For π_+, this gives 4,006 examples for `Llama-3.1-8B`, 6,797 for `Llama-3.2-1B`, and 5,566 for `Qwen3-0.6B`. For π_-, it gives 6,062, 19,051, and 9,963 examples for the same three models. Both share the same hyperparameters: learning rate 3e$-$4, 1 epoch, batch size 18, 100 warmup steps, max sequence length 2,048. LoRA: rank $r = 8$, $\alpha = 16$, dropout 0.1, targeting only `gate_proj`, `down_proj`, and `up_proj`.

cDPO fine-tuning. The final preference dataset contains 6,062 pairs for `Llama-3.1-8B`, 19,051 for `Llama-3.2-1B`, and 9,963 for `Qwen3-0.6B`, each annotated with per-token CE probabilities. Sigmoid DPO loss with $\beta = 1.0$ and no label smoothing. Training runs for 3 epochs (the setting reported as best in the original paper), with learning rate 4e$-$5, cosine decay, warmup ratio 0.1, and weight decay 0.01. Batch size 8. LoRA: rank $r = 16$, $\alpha = 32$, no dropout, targeting `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`.

Q.3 CLIFF-DPO

All Cliff-DPO variants reported in Figure 5, Table 16, and Table 18 share an identical training configuration; they differ only in which subset of cliff tokens is included in training.

Training hyperparameters. Sigmoid DPO loss with $\beta = 0.1$ and no label smoothing. Learning rate 1e$-$6 for `Llama-3.1-8B`, and 5e$-$6 for `Llama-3.2-1B` and `Qwen3-0.6B`; cosine decay, warmup ratio 0.1. Training runs for 1 epoch with an effective batch size of 64 (batch size 4, gradient accumulation 16). Gradient clipping at max norm 1.0, no weight decay. LoRA: rank $r = 32$, $\alpha = 64$, dropout 0.05, targeting `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, `down_proj`.

Q.4 Evaluation configuration

GSM8K, GSM1K, and MATH500 are evaluated using greedy decoding, with accuracy as the evaluation metric. AIME 2025 is evaluated using the model-specific sampling configurations detailed in Appendix D.2, using avg@64 as the metric. For each problem, correctness is averaged over 64 sampled responses, and the final scores are averaged across the 30 problems.

All reported scores are averaged over three independent evaluation runs. For trained models, this corresponds to one evaluation per training seed. For base models, it corresponds to three repeated evaluations of the same base model. While variations in AIME 2025 naturally stem from sampling, repeated evaluations of base models on the other benchmarks may also differ slightly despite greedy decoding, due to numerical nondeterminism introduced by continuous batching and BF16 arithmetic. Figure 5 displays 95% confidence intervals computed over these three runs. Tables 16 and 18 instead report the sample standard error, $s/\sqrt{3}$.

The *Updated tokens* column in Tables 16 and 18 reports the total number of token positions assigned a non-zero loss during fine-tuning. For DPO and cDPO, this count includes all response-token positions supervised by the preference objective. For Cliff-DPO, it includes only the selected cliff-token positions.

R Existing assets and licenses

Table 19: Existing assets used in this work and their licenses or terms of use.

Asset	Use in this work	License / Terms
`Qwen3-8B`	Evaluation model	Apache-2.0
`Qwen3-4B`	Evaluation model	Apache-2.0
`Qwen3-0.6B`	Training and evaluation model	Apache-2.0
`Llama-3.1-8B-Instruct`	Training and evaluation model	Llama 3.1 Community License
`Llama-3.2-3B-Instruct`	Evaluation model	Llama 3.2 Community License
`Llama-3.2-1B-Instruct`	Training and evaluation model	Llama 3.2 Community License
`Gemma-3-4B-it`	Evaluation model	Gemma Terms of Use
GSM8K	Training and evaluation benchmark	MIT License
GSM1K	Evaluation benchmark	MIT License
MATH500	Evaluation benchmark	MIT License
AIME 2025	Evaluation benchmark	MIT License
vLLM	Inference engine	Apache-2.0
PEFT	Fine-tuning implementation	Apache-2.0